An investigation into the applicability of the RAD methodology when applied to the development of an information system

Carter, Louise D.

An Investigation into the Applicability of the RAD Methodology when applied to the development of an Information System

Louise D. Carter

M.Sc. Thesis 2001

An Investigation into the Applicability of the RAD Methodology when applied to the development of an Information System

Louise D. Carter

M.Sc. Thesis

Department of Computer Science

University of Durham

February 2001

TABLE OF CONTENTS

TABLE OF FIGURES

ABSTRACT

Within any software-related environment many tools, techniques and SDMs (Software Development Methodologies) can be applied to help control the software development lifecycle. The aim of this thesis is to assess the applicability of the RAD (Rapid Application Development) SDM to software development when designing an MIS (Management Information System) for Philips Components Washington. Both positive and negative characteristics of the approach were examined. This thesis provides detailed conclusions on each of the identified characteristics together with more general analysis of the use of SDMs.

Research into several SDMs has been carried out in parallel with the development of the MIS, however the RAD SDM was predominantly used throughout the project. This study has been carried out through the analysis of information requirements around the Philips factory with an aim to providing one central system by which all production figures can be collated and reported upon. Prior to this study no standards for the choice of SDM were in place for software design, however previous SDMs used tended to be more traditional and structured in nature.

Through this study, additional areas of analysis have been identified in order to further investigate the RAD SDM. Recommendations have also been provided for any future software projects that Philips may undertake with the application of the RAD methodology.

ACKNOWLEDGEMENTS

The work carried out throughout this thesis is based on work carried out as part of a TCS (Teaching Company Scheme) project for Philips Components Washington in connection with the University of Durham.

I would like to acknowledge my gratitude to a number of people, without the support and assistance from whom this work could never have been completed. Dr. S. Bradley my academic supervisor for his constant guidance and support and of course his positive encouragement to remove all random apostrophes. Mr G. Brown my industrial supervisor who without his support and encouragement this project would not have been possible.

Thanks also go to the members of Philips Components IT department whose constant input provided a valuable insight into the world of manufacturing. And finally my thanks to my husband Dave, whose endless support throughout those long nights of computer crashes and lack of will to continue helped me get there in the end.

STATEMENT OF COPYRIGHT

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

The design of a system be it computerised or manual, often follows some form of structured path in order to ensure that design stays on track. Several 'System Development Methodologies' (SDMs) exist choosing the right one to suit a particular business need, can be very difficult. This often leads to methodologies being tailored to the needs of the particular environment and places more emphasis on customised solutions rather than using methodologies in their totality **[FITZGERALD, 1997]**.

"The proper place to study elephants is in the Jungle, not the zoo".

[McLEAN, 1973]

The above quote from McLean suggests the need to test SDMs in their intended environment in order to gain real world results. Throughout time, researchers have criticised the lack of empirical research on systems development in real life situations and there has been a call to gain a clearer understanding of the realities of software development **[CSTB, 1990]**.

The main objective of this project is to examine the documented characteristics of the Rapid Application Development (RAD) SDM. In order to provide empirical results on its applicability, an assessment of RAD will be made examining its suitability within the manufacturing environment of Philips Components. Analysis of other SDMs is also provided.

This chapter will provide an introduction to the project and identify the main areas of research and chapters included within this thesis. Within this study three main SDMs were researched (Traditional, RAD and Rapid Prototyping through RAD) definitions of which can be found in Chapter 3 – SDM and MIS Background.

1.2 AIMS OF THE PROJECT

Software development projects today often involve numerous revisions and undergo many changes throughout their lifecycle.

In the modern world, change is occurring at an ever-increasing rate. In the business world an organisation must adapt to this change or cease operation. The demands through lack of time that are being placed on many businesses force them to adapt to a rapidly changing environment and increase the need to implement computer systems that meet business needs in a timely fashion **[RIDGWAY-DAVIES, 1996]**.

The aim of this thesis is to detail the activities carried out during the analysis and design of a Manufacturing Information System (MIS), concentrating on how applicable the RAD SDM was within the manufacturing environment. The resulting study will provide suggestions on how RAD can be used more effectively and identify alternative approaches to analysis.

One main objective of the MIS was to support the main production activities carried out by Philips Components, helping them achieve the best possible results by providing them with accurate, more efficient data collection and manipulation. Although known as a Manufacturing Information System within the Philips environment, the ultimate objective is to provide the company with a Management Information System. The concept of MIS is explained in more detail in Chapter 3 – SDM and MIS Background.

1.3 THESIS LAYOUT

This Thesis contains the following chapters:

Chapter Title	**Description**
Chapter 2 – Company Background	This chapter examines the Project situation and the background that lead to the Project creation. The criterion by which the study was measured is also included.
Chapter 3 – SDM & MIS Background.	This chapter discusses the various SDMs examined and details their application. This chapter also examines the theory behind MIS.
Chapter 4 – The Development Process & SDM Applicability Measurements	This chapter examines the development process followed for the MIS project documenting the various stages of development. The measures by which the SDM will be assessed are also documented in this chapter.
Chapter 5 – Results	This chapter documents how RAD was applied within the project and provides an illustration of its success.
Chapter 6 – Analysis of Results	This chapter examines how RAD performed based on the results identified in Chapter 5.

Chapter Title	Description
Chapter 7 – Conclusions and Recommendations	This chapter will document the applicability of the RAD approach and assess the project against the success criteria illustrated below.

Figure 1-1 – Thesis layout

1.4 SUCCESS CRITERIA

The main success criteria for this project were:

1. To examine the perceived characteristics of the RAD approach (documented in Chapter 3).

2. To investigate the applicability of the RAD approach when designing a software solution within the Manufacturing environment.

3. To suggest ways in which the RAD approach could be tailored to overcome any identified shortfalls.

CHAPTER 2

COMPANY BACKGROUND

2.1 INTRODUCTION

This section aims to provide the reader with a basic understanding of the Philips organisation, its operations and the origin of the MIS. Firstly the Philips organisation will be examined briefly, looking at the problem area and how the idea of an MIS came to fruition. Secondly the key stakeholders within the MIS project will be examined to provide an insight into roles and responsibilities within the MIS.

2.2 PHILIPS COMPONENTS WASHINGTON

Philips Components Washington is a leading manufacturer of Deflection Units (D.U.s) for television sets. Figure 2.1 below illustrates a D.U., figure 2-2 illustrates the basic components of a television and the situation of the D.U.

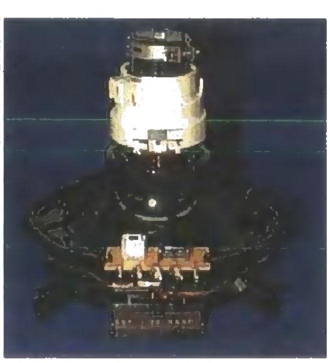


Figure 2-1 – A Deflection Unit [PHILIPS, 1998]

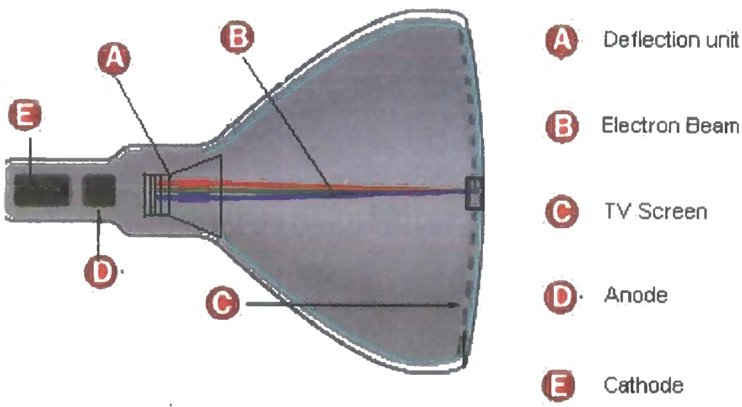


Figure 2-2 – The basic components of a television [PHILIPS, 1998]

In the above diagram it can be seen that once the electron beam is fired through the cathode ray tube (CRT) there is no way to "steer" the beam - the beam will always land in a tiny dot right in the centre of the screen. This is why a D.U. is needed. The D.U. fits on the back of the CRT and is used to deflect the electron beam from red, green and blue electron guns in the neck of the tube onto a Phosphor coating on the front of the tube. The D.U. consists of a core and copper windings. The D.U. is able to create magnetic fields inside the tube, and the electron beam responds to the fields. One set of coils creates a magnetic field that moves the electron beam vertically, while another set moves the beam horizontally. By controlling the voltages in the coils, the electron beam can be positioned at any point on the screen.

The Washington factory manufactures about 14 million D.U.s per year with around 150 product variations. Once manufactured, the D.U.s are supplied to seven internal Philips sites throughout Europe and directly to certain third party vendors. At present, the Washington factory has a workforce of 627, full and part time staff and an annual turnover of approximately £73.8M.

2.2.1 CURRENT MANUFACTURING PROCEDURE

D.U.s are manufactured on fourteen separate automated production lines around the factory Figure 2-3 below illustrates the average weekly production volume for each line in order to provide the reader with an understanding of the size of the lines.

Figure 2-4 illustrates the number of shifts working on each line within a 24-hour period. The lines AYA1,AYA2,AYA3,AYA4,AMS1,AMS2,AMS3,AMS4 and FMS3 are automated lines and the remaining lines are manual flow lines i.e. no computer automation exists on these lines.

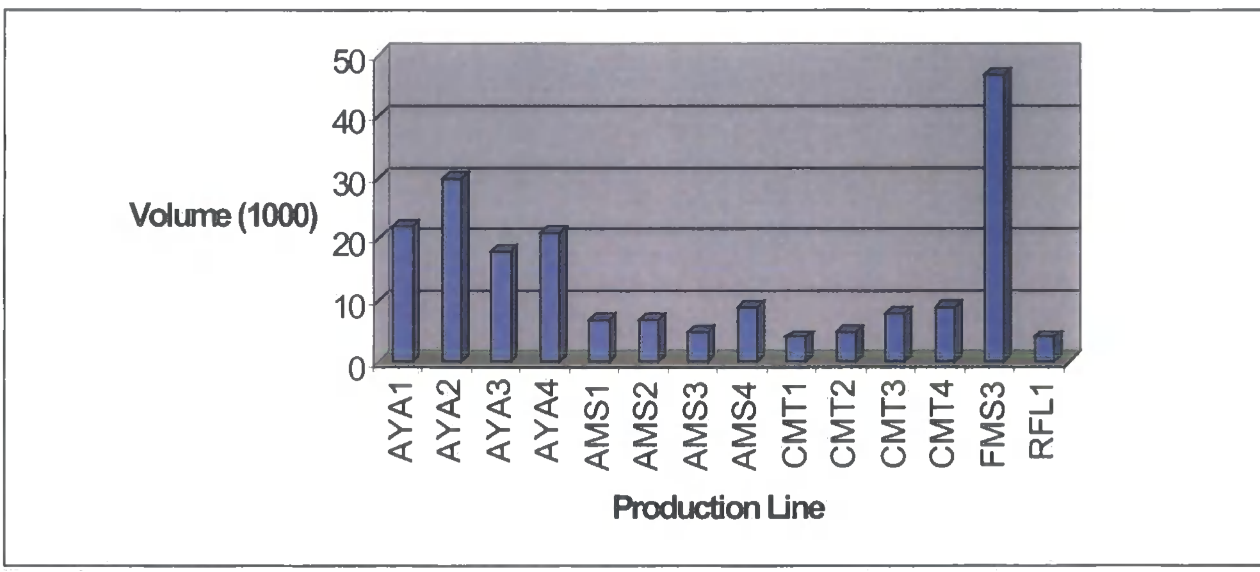


Figure 2-3 – Average weekly production line volumes

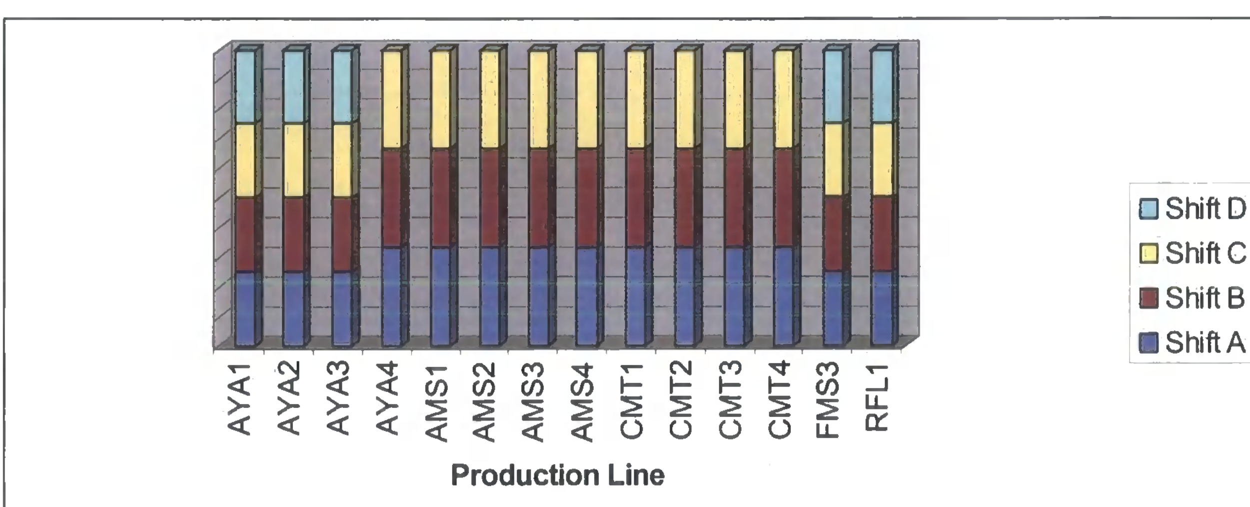


Figure 2-4 – Shift Pattern on production lines during a 24-hour period

The production process is heavily reliant on vital product information being passed between lines and departments.

Five of the fourteen production lines are controlled by Oracle programmed line controllers, these controllers feed production information to and receive data back from the production line. Oracle line controllers however are semi automated with some human input. Four of the production lines are controlled by Sybase programmed line controllers again using the transfer of production data to assemble a D.U. Sybase line controllers are fully automated with no human input to the manufacturing process. All line controller information is illustrated by the yellow area on figure 2-5. The two separate line controllers are not connected and operate as separate entities. Lines operated by line controllers are also known as 'automated lines'. The remaining three lines operate without the use of a line controller and are known as 'flow lines' or 'manual lines'. Within the flow lines, production data can be collected manually from individual PC terminals around the line.

2.2.2 Information Requirements

A brief explanation of information requirements will now be given in order to highlight the need for an MIS. In order that factory and line productivity be measured accurately, a number of different types of information are required at differing levels. Figure 2-5 below illustrates the different levels and groups of information.

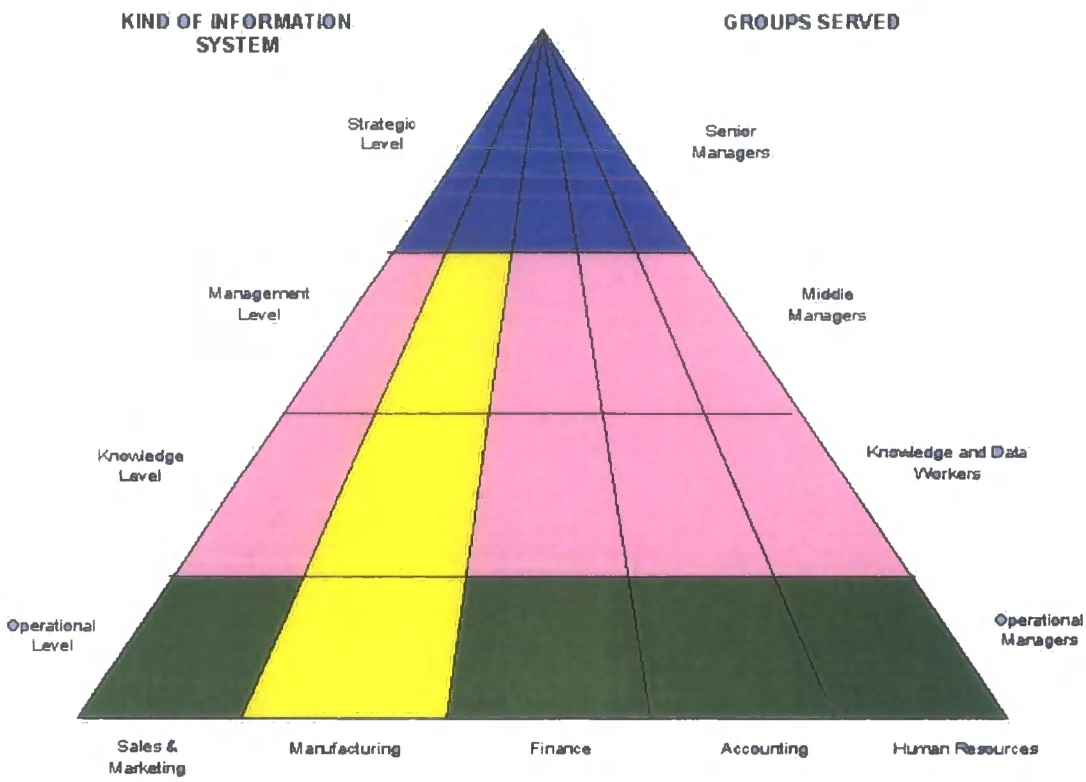


Figure 2-5 – Types of Information System [LAUDON & LAUDON, 1999].

From the above diagram it can be seen that because there are different interests, specialities, and levels in an organisation, different kinds of IS exist. Information Systems can be split into strategic, management, knowledge and operational levels each with varied information requirements. These groups can then be sub divided into further areas: sales and marketing, manufacturing, finance, accounting and human resources. The main information focus for the Philips MIS is to provide high level, strategic trend information to management allowing them to view historic production data with an aim to drive future improvements. This is illustrated in figure 2-5 by the blue area. Strategic-level systems help senior management with long term planning. However the Philips MIS also serves to provide 'Operational level' data, helping operational managers keep track of the company's day-to-day activities. This is illustrated in figure 2-5 by the green area. The main information source for the MIS system comes from the line controllers previously discussed in section 2.2.1. This is illustrated by the yellow area on figure 2-5. The MIS also has a long-term objective to address each of the sub divided areas such as sales and marketing, manufacturing and finance etc. Figure 2-6 below illustrates the current management structure within

Philips Washington. This is intended to provide the reader with an understanding of the information requirements required at each level.

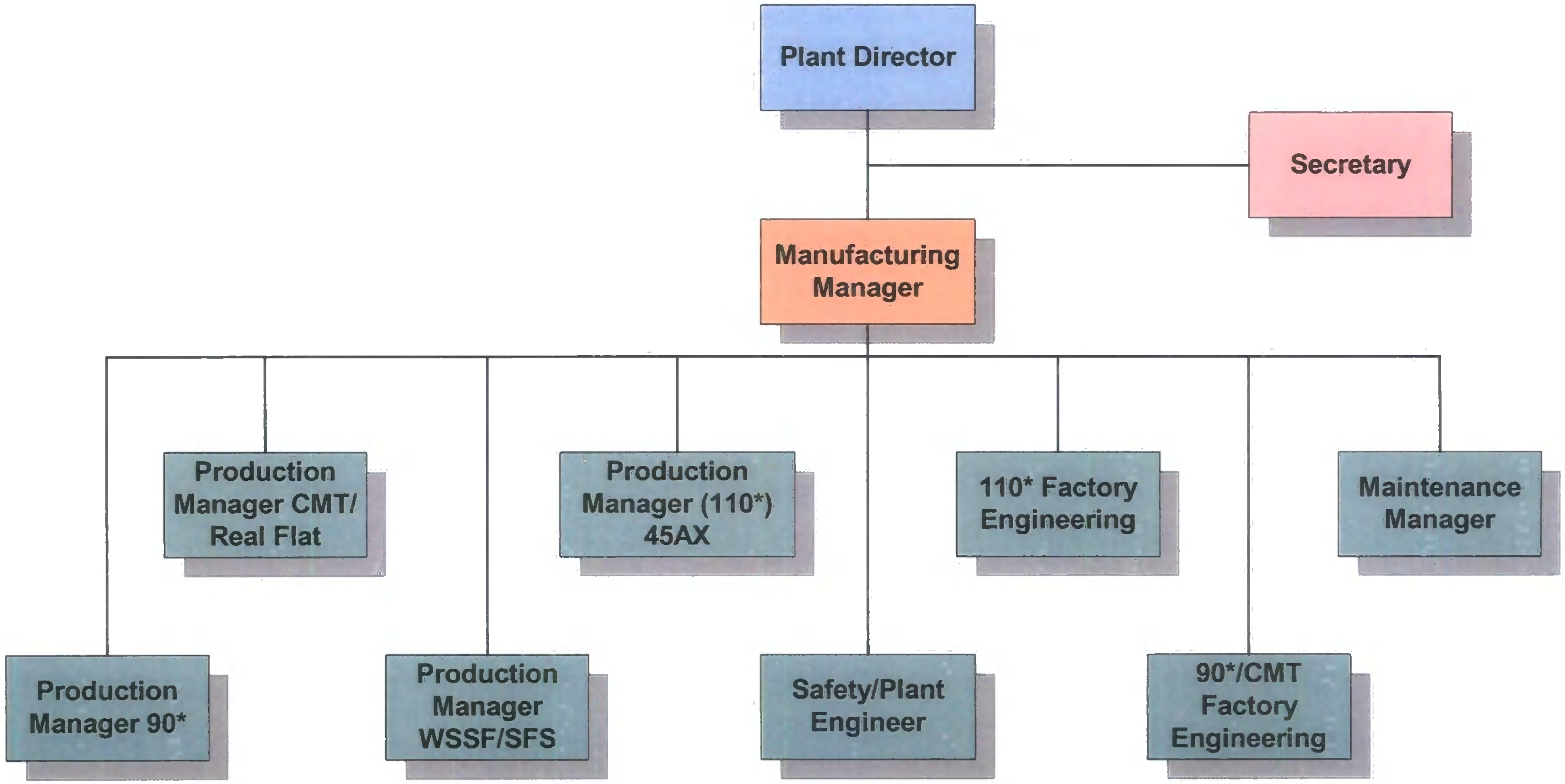


Figure 2-6 – Philips Management Structure

Within the manufacturing group several 'Key Performance' information groups or KPIs were identified at project initiation, definitions of which are provided below in figure 2-7. Each KPI concerns itself with a different aspect of factory information. KPIs were then used in various combinations to provide overall performance indicators.

KPI	Definition	Comments
Operational Equipment Efficiency (Asset Efficiency)	$\frac{\text{Output}}{\text{Max Capacity in Time Elapsed}}$	Ratio of output over maximum capacity adjusted for product mix.
Factory Efficiency	$\frac{\text{Output}}{\text{Max Capacity in Planned Run Time}}$	Ratio of output over maximum capacity of the working hours adjusted for product mix. Time for which the line was planned to be running
Performance Rate	$\frac{\text{Output}}{\text{Max Capacity in Actual Run Time}}$	Ratio of output over maximum capacity in the time it was running adjusted for product mix. Time for which the line was actually running. All times are averaged over tied cells of the same type.
First Time Yield	$\frac{\text{First Time Good}}{\text{First Time Processed}}$	First Time Yield (also known as Direct Yield)
Quality Rate	$\frac{\text{Total Good}}{\text{Total Processed}}$	Ratio of Good volume over total volume processed
Capital Utilisation	$\frac{\text{Planned Run Time}}{\text{Time Elapsed}}$	Ratio of planned run time over Elapsed time (e.g.168 hours in a week)
Availability	$\frac{\text{Actual Run Time}}{\text{Planned Run Time}}$	Ratio of running time over elapsed time (e.g.168 hours in a week) All times are averaged over tied cells of the same type.

Figure 2-7 – KPI Definitions

2.3 History of the Philips MIS

So where did the idea of the MIS originate from, why would Philips benefit from an MIS? The manufacturing process within Philips is one of the key business processes and as such the information associated with the process is key to the successful operation of the factory. In the early months of 1998 it became apparent that there were several issues related to information passage:

- Duplication of information on factory floor and Administration

- No one set of clear definitions to be applied across the entire factory
- Data collection was expensive and time consuming in terms of labour
- There was multiple ownership of spreadsheets, databases, graphs etc.

These points constitute the problem definition for the MIS project. Definition of the problem highlighted an increased need for more efficient, timelier data passage around the factory. The Philips factory could no longer rely upon estimates, management needed more accurate information. In June 1998 the MIS project began by examining the information requirements on the factory floor.

2.4 OBJECTIVES OF THE PHILIPS MIS

Within the Philips factory, information is required at two main levels, operational and managerial/strategic. Operational information (illustrated by the green area in figure 2-5) involves the day to day running of a manufacturing line, whist managerial requirements (blue area in figure 2-5) mainly involve ensuring that the manufacturing process operates efficiently, thus enabling management to make timely decisions. The vision statement for the MIS is as follows:

"To provide a standard structure for gathering, calculating and formatting of data across all manufacturing lines within the factory."

[PHILIPS, 1998]

The main short-term objectives of the MIS were to provide structured data, provide single sources of data and definitions and set a strong foundation for defined KPIs. The main long-term objective of the MIS was to automate data wherever possible.

2.5 MIS STAKEHOLDERS

As mentioned previously, information within the MIS is required at two levels, managerial (M) and operational (O). Before deciding on a design for the MIS, examination of key stakeholders in the project was carried out. A Stakeholder can be defined as any person or department that has a direct connection with the MIS system, a stakeholder can be either internal or external to the organisation. The Stakeholder diagram shown in figure 2-8 illustrates the stakeholders involved with the MIS, the information that they require and the flow of information between departments. This diagram also helps to highlight the scope of the MIS project.

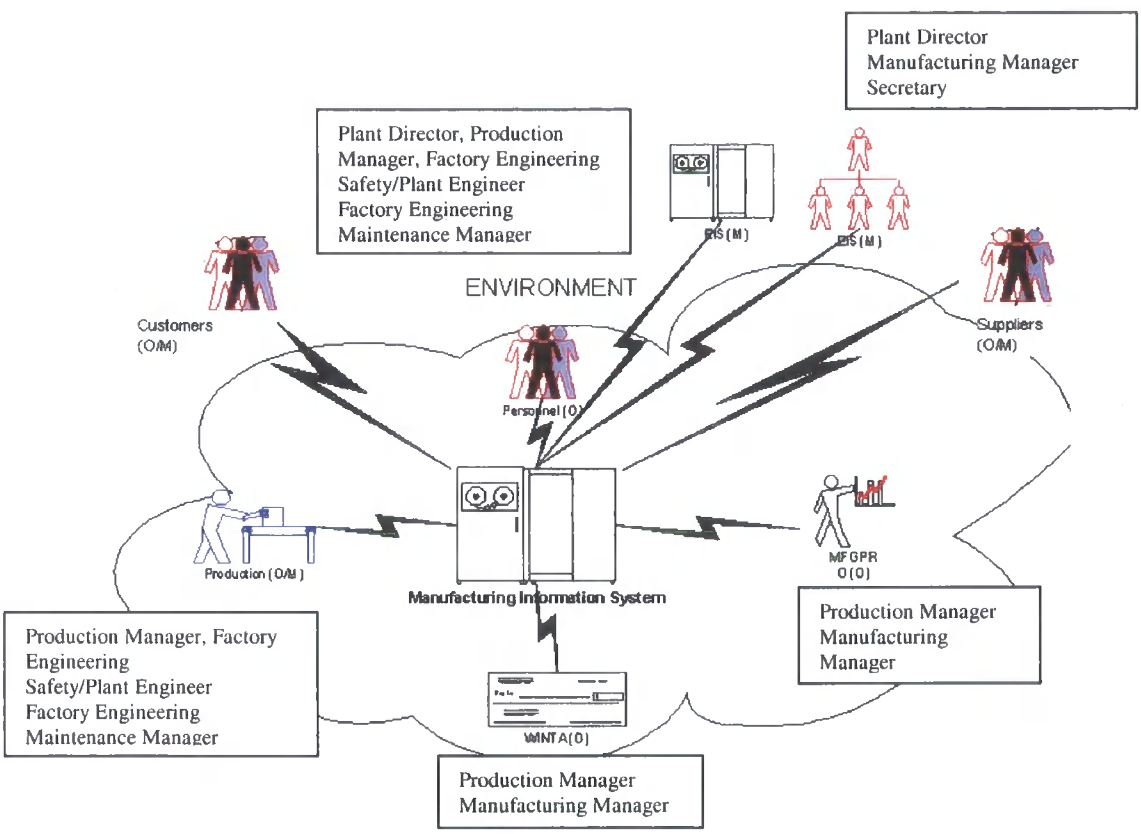


Figure 2-8 - MIS Stakeholder Diagram

Stakeholder Diagram Key

WINTA – Windows Time and Attendance System – Clocking in System
EIS – Executive Information System – Corporate Information System
MFGPRO – Manufacturing Professional – Enterprise Resource Planning System
(O) – Operational Information
(M) – Management Information

2.5.1 STAKEHOLDER DESCRIPTIONS

Production - forms the main stakeholder within the MIS system. The information processed within production is a combination of both operational and managerial (strategic) information. Production provides the MIS with its main data source, taking information from a number of systems, processing this information and then passing this to the MIS. Information at this level is both manually collected and automated and contains both technical and personnel (payroll) content. With reference to figure 2-5 production is covered by the 'manufacturing' band.

Personnel - acts as both a data source and a data sink (receives data) for the MIS. Information within personnel is of an operational nature and consists of content such as labour hours, employees time on line, holidays etc. This information is provided through production and processed within the MIS. MIS will also request information from personnel, information such as employee work codes, contracted work hours etc. With reference to figure 2-5 personnel is covered by the 'human resources' band.

MFGPRO - the main resource planning system used within Philips Washington. This system is used to plan factory-manufacturing capability based on customer demand and available resources. MFGPRO will use MIS production information such as product cycle times, line efficiency, labour availability etc., as an input to the planning process. Information at this level is mainly operational as MFGPRO relies upon accurate day-to-day data. With reference to figure 2-5 MFGPRO is a combination of manufacturing, finance, accountancy and human resources.

WINTA - Philips Washington's 'clocking in' system that provides personnel with information required for payroll, roll calls, employee hours etc. WINTA is used as both a data source and a data sink for MIS, the MIS system will provide WINTA with operational information regarding employee hours on line, employee holidays etc. WINTA will provide MIS with information such as absence codes and operator hours within the factory. With reference to figure 2-5 WINTA is covered by a combination of finance and human resources.

EIS - Philips corporate Executive Information System. EIS differs from the MIS in that it is solely concerned with top-level strategic data whereas the MIS uses both strategic and operational data. It uses the output from the MIS to provide strategic planning decisions. EIS receives data from all Philips factories world-wide and is used as a global management tool providing Philips management with managerial (strategic) information regarding the efficiency and operation of each Philips factory. Interaction with EIS involves both communication with computer systems and management teams. With reference to figure 2-5 EIS is covered by a combination of all sections but at a high strategic level.

Customers - Within Philips Components Washington customers can be both internal and external; ultimately a customer is someone who will purchase a product from the Washington factory. One example of an internal customer could be a Philips TV Tube manufacturer that requires D.Us from Washington in order to complete production of their TV tubes. External customers could range from major TV manufacturers to computer monitor manufacturers. In both cases information is passed to and received from the MIS in many different forms. A particular customer may require production figures for a specific make of D.U whilst another customer may wish to enquire about the performance of a D.U.

Suppliers - Suppliers in the context of Philips Components can take many forms. One example could be suppliers of the raw materials used to make a D.U. another example could be the delivery company used to ensure the D.U. reaches its customer. In the majority of the cases suppliers will provide an information feed into the MIS system however some suppliers may also require information from the MIS. An example of this could be a copper supplier enquiring about the cycle time that the manufacturing process takes to wind one coil of their copper. The MIS could provide suppliers with benchmark information.

2.6 SUMMARY

This chapter has helped to highlight the problem situation (illustrated in section 2-3) to which the RAD SDM was applied and the key stakeholders that will be affected by the scope of the study. This chapter has also helped to identify the current situation within Philips illustrating the need for an information system.

This chapter also highlights the need for an SDM that addresses not only technical, complex issues but also the human and organisational aspects of an information system within a manufacturing environment.

CHAPTER 3

SDM AND MIS BACKGROUND

3.1 INTRODUCTION

This chapter will illustrate the main SDMs that were examined throughout the development of the Philips MIS and provide an explanation of the origin of each. It is important to note that research into the various SDMs was carried out in parallel with the MIS development and hence some of the documented techniques may not have been fully applied within this project. The latter section of this chapter will examine information systems and their origins. Although not directly related to the SDMs applied throughout the project, this section has been included to provide the reader with an understanding of the objectives of the MIS.

3.2 REQUIREMENTS OF A SYSTEMS DESIGN METHODOLOGY

This section will explain the need for an SDM when analysing a problem situation.

When preparing to analyse any situation, the process is aided by using some form of analysis methodology. An SDM provides an analyst with a set of guidelines to follow to ensure that all aspects of design are covered.

Avison and Fitzgerald **[AVISON & FITZGERALD, 1995]**, in their work with information systems provide the following definition of an SDM:

A collection of procedures, techniques, tools, and documentation aids which will help the systems developers in their efforts to implement a new information system.

Throughout this thesis the terms ***SDM*** or ***methodology*** are used in place of the above definition.

According to Maclean and Stepney **[MACLEAN, STEPNEY 1994]**, an SDM provides analysis of any Information Technology (IT) system by dividing development into four key areas. The chosen methodology should be able to support all four of these areas. It is important to note that the sequence in which these analysis tasks need to be carried out is highly determined by the situation. The four key areas of analysis are identified in the following table – Figure 3-1.

Analysis Task	**Description**
Scope	Identifying the areas and activities within an organisation that will be studied and ruling out areas that are not of interest.
Problem Definition	Examining what current system problems exist and highlighting possible contributors to the problem.
Proposed Solutions	Determining suitable solutions for the identified problems.
Recommended IT support and backup	Identify what IT components are required to help solve the problem.

Figure 3-1 - Stages of IT analysis

Edward Yourdon in his book 'Managing the System Lifecycle' refers to a quote by Tom De Marco stating that 'You can't have a methodology without methods.' You can't hope to organise, manage and control a software project unless you can describe to the designers, technicians and programmers – *what* you expect them to be doing during the various stages of the project. Also information needs to be supplied on *what* kind of technical products are expected **[YOURDON, 1982]**.

Penny Kendal in her book Introduction to Systems Analysis and Design **[KENDAL, 1989]** states that one main characteristic of any SDM is the ability of the methodology to help developers build maintainable systems. She explains that systems analysis and design is still a rather subjective science involving people. SDMs are tools to be used as and when required and need not necessarily be used religiously.

Smiles **[SMILES, 2000]**, states that regardless of the SDM used, all methodologies have one thing in common, that at some point in the lifecycle, software has to be tested.

3.3 Software Development Lifecycles

Regardless of the SDM chosen, each will follow some form of prescribed 'lifecycle'. The various activities that are undertaken when developing software are commonly modelled as a Software Development Lifecycle or SDLC. The SDLC begins with the identification of a requirement for software and ends with the formal verification of the developed software against the documented requirements **[SMILES, 2000]**. There are a number of different models for SDLC; the more commonly used ones are explained in figure 3-2 below.

SDLC Type	**Identifying Characteristic**
Waterfall	The main characteristic of the 'waterfall' SDLC is its ***sequential*** nature. Within this SDLC development will progress through a number of well-defined phases. Once all phases have been completed a final product will be produced. SSADM is a good example of this.
Progressive	This type of SDLC provides 'interim' deliveries of software. Each individual phase of development will follow its own SDLC. The number of phases depends on development **[DAY, 1999]**.
Iterative	An iterative SDLC does not attempt to start with a full specification of requirements. Instead development begins by specifying and implementing just part of the software. This is then reviewed in order to identify further requirements. The process is then repeated, producing a new version of the software for each cycle of the model **[SMILES, 2000]**. Iterative SDLCs provide a method of detecting errors early in the development lifecycle. Generally in systems design, the later on in the lifecycle that errors are detected, the more expensive it is to rework a solution and more time is taken in order to correct the errors **[MARTIN, 1991]**. Figure 3-3 below indicates the relative cost of reducing errors through the development

SDLC Type	Identifying Characteristic
	lifecycle and illustrates the main focus of the RAD approach. RAD and Rapid prototyping are good examples of this type of SDLC.
Spiral	Development will follow a software spiral. At the end of each complete spiral, a decision will be made, either to continue through another cycle or, if the risks are too great, to leave the software as the final product. Spiral SDLCs also help to provide early detection of errors. With each loop through the spiral, more complete versions of the software are produced **[DAY, 1999]**. RAD is a good example of this

Figure 3-2 – SDLC Descriptions

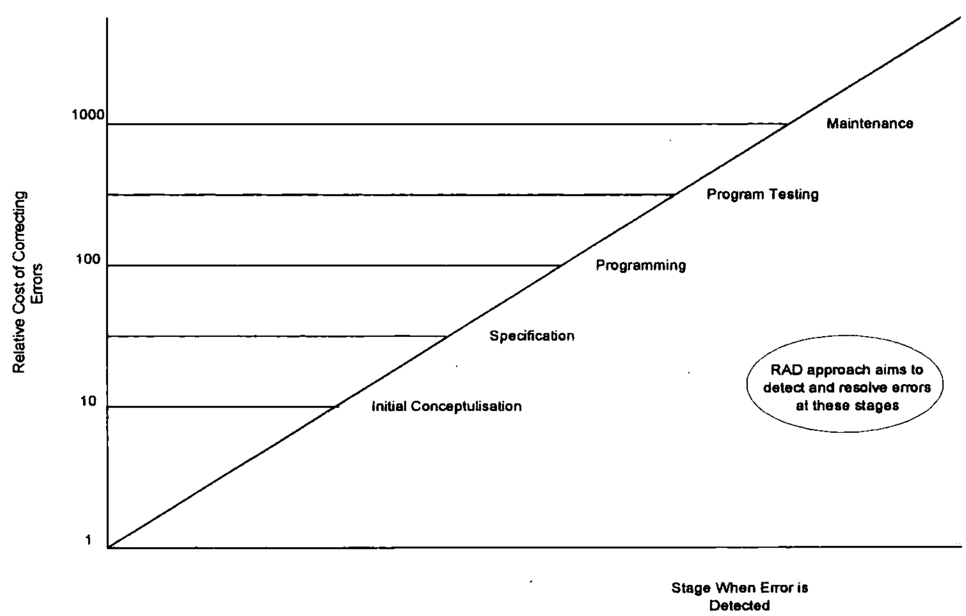


Figure 3-3 - The Relative Cost of Correcting Errors Through the Development Lifecycle [UCD, 1997].

3.4 TYPES OF SDM

There are many differing types of SDM that can be used in software development. The following SDMs were examined in parallel with the development of the MIS:

- Structured Systems Analysis Design Methodology (SSADM) – Waterfall lifecycle
- Rapid Prototyping
- Boehm's Spiral Development Process
- Soft Systems Methodology (SSM)
- RAD

This section will provide examples of each SDM, explain their origins and demonstrate perceived advantages/disadvantages.

3.4.1 SSADM

SSADM typically follows a waterfall lifecycle due to its sequential nature and step-by-step approach. It was developed as a systematic, and obvious, sequence of steps beginning with requirements and progressing through to maintenance **[DAY, 1999]**. The waterfall development lifecycle can take many forms. This section will provide graphical examples of these followed with a description of the stages involved. One example of a waterfall SDM is illustrated in Figure 3-4 below **[KENDAL, 1989]**.

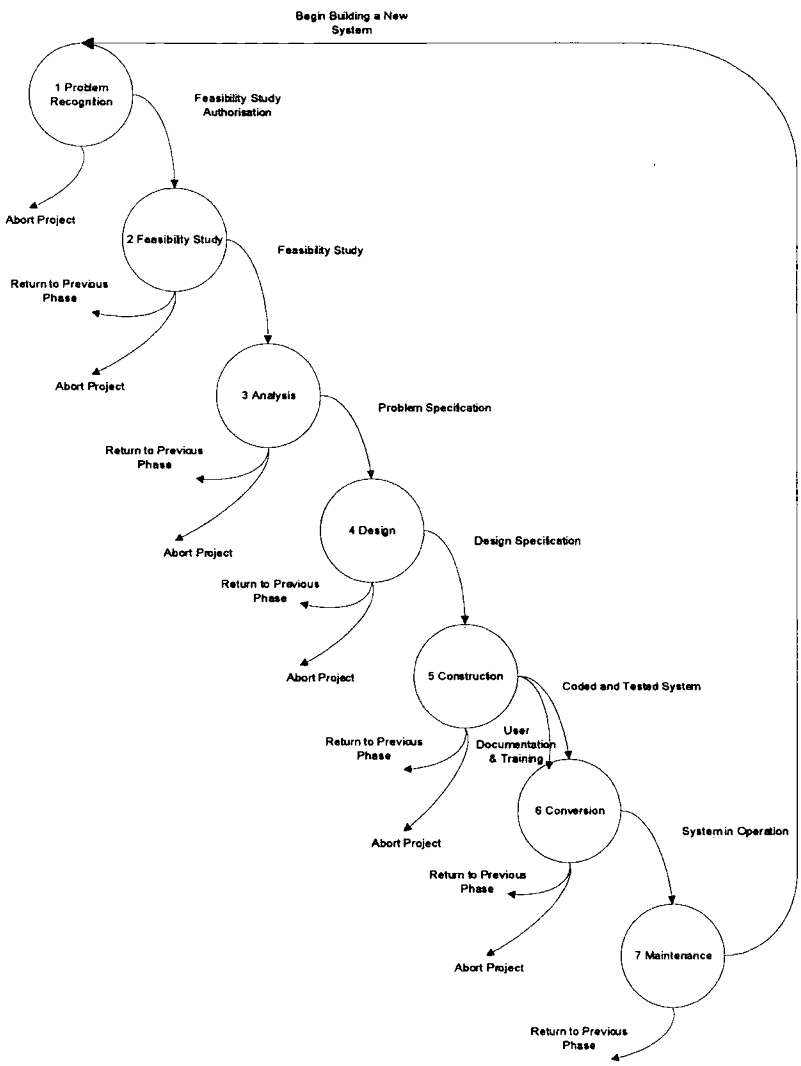


Figure 3-4 – A Typical Waterfall SDM [KENDAL, 1989]

3.4.1.1 STAGES OF A WATERFALL SDM

The stages of the waterfall SDM illustrated above will now be explained in figure 3-5 below.

Stage of Lifecycle	Characteristics
Problem Recognition/Terms Of Reference (TOR)	• Completed well before the project commences. • Details the aims and objectives of the project and provides some background on the current situation.
Feasibility Study	• Examines if the TOR are achievable and if the project should go ahead. • Should be completed within one month. • Once the feasibility study is complete the analyst must determine if the required system is technically, humanly, and economically feasible for the company.
Analysis	• Involves studying the current system (if one exists) to find out exactly where the problems lie. • Analysis techniques could include interviews, reading existing documentation and examining current procedures. • Main output from this stage will be a problem specification defining the requirements of the proposed system.
Design	• Information gathered throughout the previous stages will be brought together and used as a basis for the design of the final system. • If accepted, the design is implemented and tested, and once all parties are satisfied the system is released as operational. • Requirements are often frozen at this stage so that the exact specifications for the system can be documented **[DAY, 1999]**. • Information contained within the problem specification will now be translated into plans for a series of computer programs that will perform the functionality required by the new system.
Construction	• At this stage the system design is translated into a physical, operational system. The more physical aspects of the project are carried out such as the writing of code, code testing, the purchase of hardware and software, system set up, user testing and user training.

Stage of Lifecycle	Characteristics
Conversion	• Old, existing systems are converted into the new systems. • May be carried out gradually where part of the new system is implemented one month and the following month another part is implemented. • Alternatively conversion can be carried out using a 'big-bang' method where the old system is completely switched off and replaced totally by the new system.
Maintenance	• Required changes or modifications are made to the system once it is operational. • A large proportion of programmer effort typically 50% – 70% is concentrated on this stage **[KENDAL, 1989]**.

Figure 3-5 – stages of a waterfall SDM

3.4.1.2 Application of the Waterfall Methodology

It can be seen with reference to the above that typically, the waterfall approach suggests that each phase of the development process should be carried out and fully completed before moving onto the next stage. The approach demonstrated by Kendal suggests that previous phases must be revisited if problems occur at any stage; this ensures that issues are not carried over into future phases. Generally, IT staff carry out the earlier stages of the lifecycle, user involvement usually occurs towards the end of the analysis phase or after the implementation phase. Once a solution has been developed it will be tested and implemented into its intended environment.

3.4.1.3 Advantages/Disadvantages of the Waterfall SDM

Advantages	Disadvantages
• Meaningful paper models of the system and high visibility of analysis, design and programming effort.	• Maintenance work is often high due to 'Frozen' requirements definitions **[DE MONTFORT UNIVERSITY, 1999]**

Advantages	Disadvantages
• Provides objective criteria for measuring and determining quality. Due to the formal nature of this SDM, uniformity and conciseness is inherent throughout the lifecycle, leading to a more controlled, higher quality deliverable.	• The scope stage tends not to take adequate account of the way in which the system must fit into the overall business systems environment. This leads to constant redefinition of scope **[HOWARD, 1997]**.
• Well defined documentation **[FOWLER, 2000]**	• Although well defined, the analysis documentation in waterfall lifecycles is not sufficiently defined enough to be used as a basis for direct coding in the development stage. This leads to rework in the coding stage and hence a delay to the project.
• Progress can be reviewed at the end of each stage **[LAWS, 1996]**	• Too little user contact. In most waterfall SDMs, the user does not get to see the system until it is coded and implemented as a final system. Users often are unsure of what they are getting **[FOWLER, 2000]**
	• Very Little Iteration. The design has to be right first time **[FOWLER, 2000]**
	• Failure to meet the needs of management. Waterfall SDMs tend to ignore the management/strategic levels of an IS. They are mainly concerned with day-to-day operational processing **[LAWS, 1996]**.

Figure 3-6 – Advantages/Disadvantages of the Waterfall SDM

3.4.1.4 Summary of The Waterfall SDM

The waterfall SDM is said to provide the analyst with a highly structured, formal approach to systems analysis. This SDM provides well-defined deliverables at each stage of the process, providing a thorough investigation of the situation under analysis. However, this highly rigid approach with its emphasis on sequential activities, pre-defined detailed deliverables and separation of users from IT specialists has been heavily criticised for delivering systems that are over budget, over schedule and not what the user actually requires **[AVISON & FITZGERALD, 1995]**.

In their study of the waterfall SDM, Kinmond and Bharti state that this approach can lead to many failings when applied to situations that involve short timescales and rapidly changing requirements **[KINMOND & BHARTI, 2000]**. This stresses the need for a more flexible approach to systems development.

3.4.2 Boehm's Spiral Development Process Model

Figure 3-7 below illustrates the original Spiral Model published by Boehm **[BOEHM, 2000]**. The following overview definition is provided by Boehm to describe the essence of the spiral model:

The spiral development model is a risk-driven process model generator. It is used to guide multi-stakeholder concurrent engineering of software intensive systems. It has two main distinguishing features. One is an iterative approach for incrementally growing a systems degree of definition and implementation while decreasing its degree of risk. The other is a set of anchor point milestones for ensuring stakeholder commitment to feasible and mutually satisfactory system solutions **[BOEHM, 2000]**.

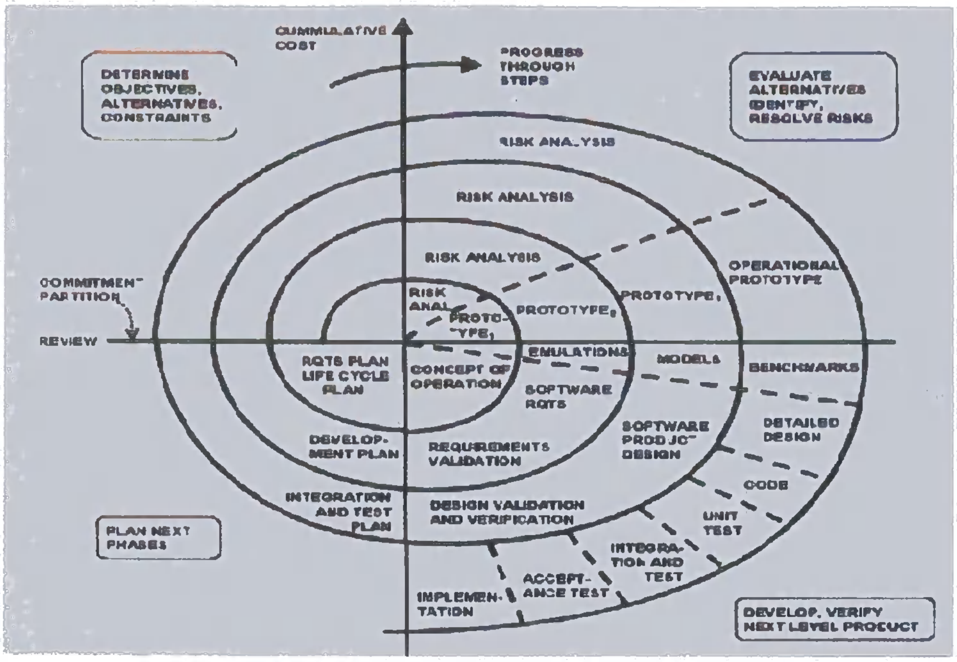


Figure 3-7 – Boehm's Spiral Model [BOEHM, 2000]

3.4.2.1 Stages of the Spiral Model

As can be seen from the above illustration, the model comprises of several iterations through four main quadrants. Figure 3-8 below describes the stages of the Spiral Model.

Stage of Lifecycle	Characteristics
Determine Objectives	• User presents the analyst with a list of requirements. A full requirements analysis will be carried out.
Evaluate Alternatives, Identify and Resolve Risks	• An assessment is made on the risk level of the project. Risks are situations or possible events that can cause a project to fail to meet its goals. Risks are prioritised by importance and then measured by the likelihood of the risk happening. • Key project stakeholders must participate concurrently in reviewing risks and choosing the projects process model accordingly.

Stage of Lifecycle	Characteristics
Develop, Verify next level product	• At this stage requirements are prototyped and validated against initial objectives. If the user is satisfied a detailed design is carried out this design is then translated into code and tested.
Plan Next Phases	• This stage consists of the planning stages of the project. Once requirements have been agreed and prototypes produced, plans will be devised for the next stages of the project.

Figure 3-8 – Stages of the Spiral Model

Boehm continues to state that a process model answers two main questions: 1) what should be done next? And 2) for how long should it continue? Using the spiral model, the answers to these questions are driven purely by risk considerations and vary from project to project. Each choice of answer generates a different process model.

3.4.2.2 Advantages/Disadvantages of The Spiral Model

Advantages	Disadvantages
• Risk is eliminated through each iteration of the process • Lowers development costs by early identification of non viable options • Anchor point milestones are used to assess project completion. Only when a milestone has been satisfied will the next stage of the project take place. • Accommodates preparation for lifecycle evolution, growth and changes of software product **[BOEHM, 1989]**. • Provides a framework for integrated hardware-software system development **[BOEHM, 1989]**.	• Model can be misleading and be used in an iterative 'waterfall' manner • Many people tend to use the model as a single spiral sequence – trying to develop all functionality in one go. • Involvement of key stakeholders in the development process can tend to be overlooked. • Heavy Reliance on risk-assessment expertise **[BOEHM, 1989]**. • Risk driven specifications can be people dependant **[BOEHM, 1989]**.

Figure 3-9 – Advantages/Disadvantages of the Spiral SDM

3.4.2.3 Summary of the Spiral SDM

From the above it can be seen that if the spiral model is used in the prescribed manner it can provide a high quality product through the continuous elimination of non-viable solutions. According to Boehm **[BOEHM, 1989]**, the risk driven nature of the spiral model is adaptable to the full range of software project situations due to its progressive nature, probably more so than approaches such as the waterfall model. This risk driven model uses concurrent rather than sequential assessment of feasibility and hence only workable solutions will be produced. However if the model is used incorrectly development will follow a sequential pattern and some risks may be overlooked.

3.4.3 Rapid Prototyping

This SDM was developed as an extension of the waterfall SDM, recognizing the fact that it is easier to understand a problem (and hence easier to produce a more complete solution) once analysis and design have been carried out and a prototype has been produced **[DAY, 1999]**. This SDM is a good example of an iterative SDLC. Figure 3-10 below illustrates a typical Prototyping SDM.

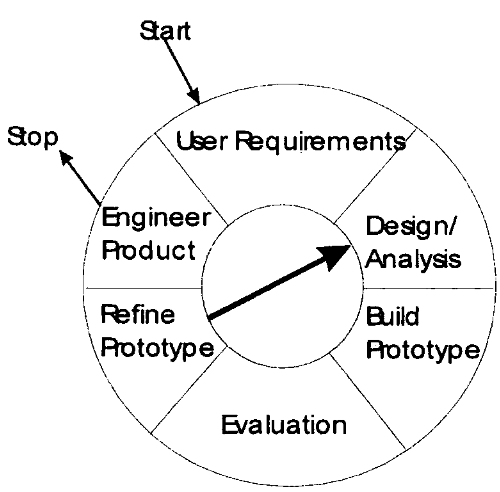


Figure 3-10 – A Prototyping SDM [DAY, 1999].

Figure 3-11 below illustrates the stages of the prototype SDM shown above.

Stage of Lifecycle	**Characteristics**
User Requirements	• Gathering of requirements usually by a process of the developer and user defining the overall objective of the system.
Design/Analysis	• A quick initial Design covering the objectives defined in the previous stage.
Build Prototype	• Construction of the prototype, which is then evaluated by the user.
Evaluation	• The user critically evaluates the prototype against the defined objectives.
Refine Prototype	• The prototype can be refined based on the users evaluation.
Engineer Product	• Iteration continues until the user is satisfied, at this stage the final product can be produced.

Figure 3-11 – Stages of the Prototype SDM [DAY, 1999].

Waterfall SDMs can use prototyping methods that supply a visualisation of the proposed system but little functionality. Rapid prototyping aims to provide a working prototype that can be used to assess both technical and non-technical feasibility. Prototyping is particularly beneficial in situations where the application is not clearly defined. Andy Laws states that the use of a prototype provides potential for early amendments to weaknesses in the designed system and, in extreme circumstances abandonment of a project **[LAWS, 1996]**. Initially within the computing environment, prototyping became the main focus of hardware designers, software design was seen more as an art and hence prototyping was seldom used. Many software engineers are now realising that prototyping is just as essential in software as it is in hardware and can be used to provide a dynamic, visual model of an application. Utilising prototype tools within software design can help to stimulate user interface design and modelling **[ISENSEE & RUDD, 1996]**.

Rapid Prototyping is not however limited purely to the software domain, many manufacturing industries also use this method. Companies such as Ford, Rolls Royce and General Motors to name a few have been using Rapid Prototyping for many years. According to a report produced by Foundry OnLine **[FOUNDRY, 2000]**, within the manufacturing industry Rapid Prototyping is used to provide many benefits, these include; increase in sales, reduction of cost, expansion of product range and fast product development.

3.4.3.1 Advantages/Disadvantages of prototyping

Advantages	**Disadvantages**
• Provides cost savings by dramatically reducing total life cycle costs.	• High levels of user support. The knowledge and support of end users is crucial to the success of the project.
• Provides the ability to refine program design quickly without attempting to code a design that doesn't meet the users requirements **[ISENSEE & RUDD, 1996]**.	• Expectation setting. The objectives of the prototype must be clearly conveyed at the start of the project. Users and management must be fully aware of what the prototype is demonstrating.
• Increased Quality. Rapid prototyping provides the user with access to a working prototype, testing can begin early and the final end product will contain fewer functional defects as a result, thus providing a higher quality deliverable **[ISENSEE & RUDD, 1996]**.	• Ending the Prototype stage. Since prototyping is so iterative in nature, a clear 'cut off' point is necessary in order that a final product be implemented. The 'Timebox' method described in section 3.3.6 could be used to overcome this problem **[LAWS, 1996]**.
• Provides early demonstration of progress **[DAY, 1999]**.	• Within the manufacturing industry changes to design can incur very large costs. Some design changes may even lead to complete re-manufacture **[FOUNDRY, 2000]**.

Advantages	Disadvantages
• Provides control over maintenance work. E.g. prototyping aims to eliminate the misinterpretation of requirements and hence maintenance work is reduced later in the lifecycle **[KINMOND & BHARTI, 2000]**.	
• Provides early demonstration of project feasibility. If any requirements cannot be supported this will be highlighted and amendments made rather than continuing development of the wrong product.	

Figure 3-12 – Advantages/ Disadvantages of Prototyping

3.4.3.2 SUMMARY OF RAPID PROTOTYPING

In summary it can be seen that the use of rapid prototyping can help to cut both project timescales and overall costs and help to increase customer satisfaction. However prototypes must be used in a controlled manner. Without structure, prototyping can be very destructive to product development with one of the key points to consider being good expectation setting on behalf of the prototypers, developers and end users.

3.4.4 THE SOFT SYSTEMS METHODOLOGY (SSM)

Another SDM to be examined was the SSM by Peter Checkland **[CHECKLAND & CHICHESTER, 1981]**. SSM is, in reality, a set of methodologies. Each methodology is represented by a set of ideas (concepts) structured in such a way that their use is appropriate to the situation being analysed. According to Wilson **[WILSON, 1999]**, each situation is unique and hence the methodology must be tailored to fit the situation and also the style of the analyst using it. SSM is a good example of an iterative SDLC.

3.4.4.1 Stages of SSM

SSM is based on what is known as the seven-stage model, this process was first identified in Checkland's early work in his book Systems Thinking, Systems Practice **[CHECKLAND & CHICHESTER, 1981]**. The seven-stage model has been in use now for over two decades and Checkland together with many other practitioners has revised the model, figure 3-13 below illustrates the original seven-stage model.

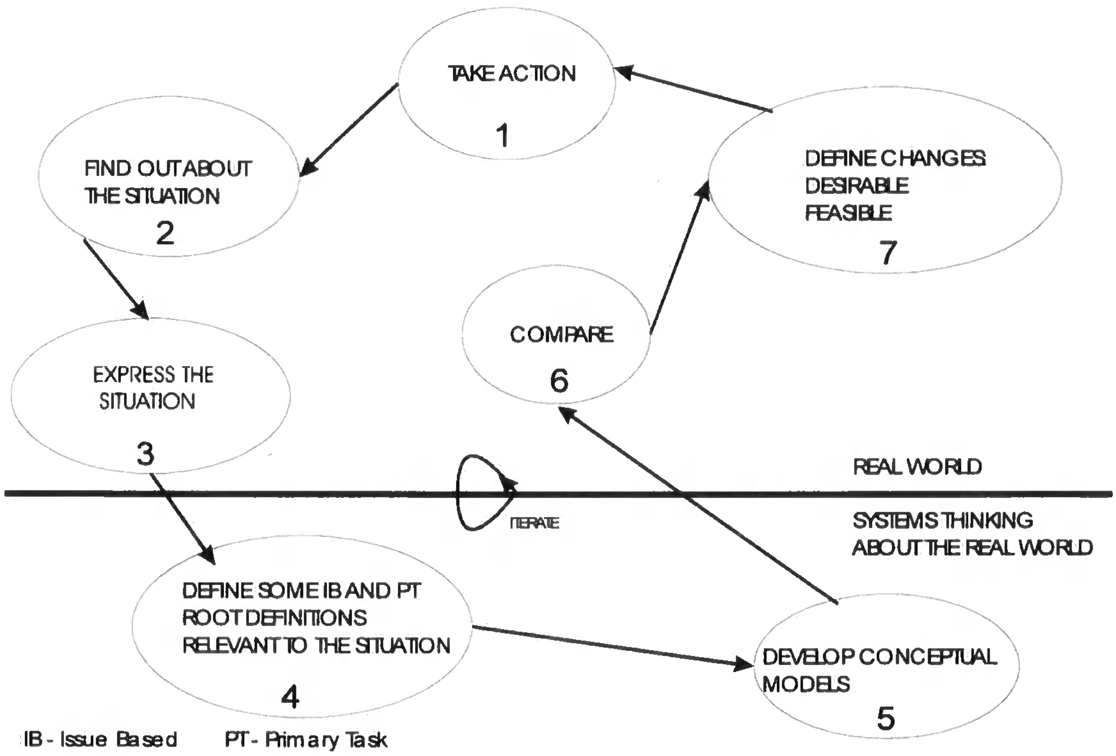


Figure 3-13 – The Checkland Methodology [WILSON, 1992]

Figure 3-14 below describes each of the above stages.

Stage of Lifecycle	Characteristics
Take Action	• This initial stage consists of identifying a problem situation considered problematic. This stage takes place in the Real World.
Find out about the situation	• This is where the situation is examined more thoroughly.

Stage of Lifecycle	Characteristics
Express the situation	• At this stage the problem situation will be defined in real terms or plain language. This stage takes place in the Real World.
Define some issue based and primary task root definitions relevant to the situation.	• This stage takes place in the conceptual world; a root definition is a base definition of the system to be designed **[WILSON, 1992]**.
Develop Conceptual Models	• A Conceptual Model is a diagrammatic representation of the activities involved within the Root Definition described above **[WILSON, 1992]**.
Compare	• At this stage the model moves back into the Real World and the CM produced is compared against the Real World situation.
Define Changes	• The output from the comparison stage leads to suggestions for change to the Real World situation; changes should be both desirable and feasible **[WILSON, 1992]**.

Figure 3-14 – Stages of the Seven-Stage SSM Model

SSM is a very 'people' orientated approach. Checkland states that since each category of participants – designers, managers, users, or other beneficiaries – may have different views of the organisations main goals or activities, many different viewpoints will be elicited during an SSM investigation **[CHECKLAND, 1990]**. It is the elicitation of these viewpoints that lead to the development of a system in terms of some purposeful human activity.

SSM uses a basic schema to categorize a situation; this schema can be expressed in terms of a mnemonic "CATWOE", which Checkland **[CHECKLAND, 1990]** explains as follows:

C – Customers the beneficiaries or the victims of T

A – Actors those who would perform T

T – Transformation process, the conversion of input to output

W – Weltanschauungen the worldview which makes T meaningful in context

O – Owners those who could stop T

E – Environment elements outside the system which it takes as a given

Viewed as a set of techniques, SSM relies upon the analyst's ability to identify various CATWOE models, and to use them the construct root definitions. The ability to visualize various aspects of situations, that is, to express complex relationships in diagrammatic or iconic form is crucial to the SSM process. The resulting output is what SSM practitioners call rich pictures. The rich picture provides the analyst with a device to help them delineate relationships, isolate roles of individuals, identify conflicts or to communicate about other aspects of a situation **[CHECKLAND, 1990]**.

3.4.4.2 Advantages/Disadvantages of SSM

Advantages	**Disadvantages**
• Using the seven stage model viewpoints can be gathered from a wide range of communities	• The analyst must have the skill to diagrammatically represent the proposed system
• Complex or 'open ended' situations can be modelled effectively **[WILSON, 1999]**.	• The analyst must be able to make a clear distinction between real world and conceptual activities
• SSM seeks to elicit the real business of an organisation as viewed through the eyes of its users **[CHECKLAND, 1990]**	• The analysts own perception of the real world may affect the outcome objectivity is crucial.
	• Rich pictures can often become too 'messy' for official production and can often contain speculation about the situation and its participants **[WILSON, 1999]**.
	• The temptation exists for the analyst to

Advantages	Disadvantages
	model what is currently happening and not what is intended for the proposed system

Figure 3-15 – Advantages/Disadvantages of SSM

3.4.4.3 Summary of SSM

SSM attempts to foster learning and appreciation of a problem situation between a group of stakeholders rather than set out to solve a predefined problem. SSM is ideally suited to situations where problem definitions are difficult and the situation is ill defined or very complex. However since SSM is a very 'perception' based SDM, it is easy for personal viewpoints and conflicts to affect the final outcome. Therefore it is crucial that those involved remain objective throughout.

3.4.5 Rapid Application Development Methodology (RAD)

This section will examine the Rapid Application Development Methodology or RAD SDM. RAD follows an iterative lifecycle model; it does not attempt to start with a full specification of requirements. Instead, development begins by specifying and implementing just part of the software, which can then be reviewed in order to further requirements. This process is then repeated, producing a new version of software for each cycle of the model **[SMILES, 2000]**. Using this iterative approach to development, the RAD SDM is quoted to provide implementation of systems within 90 days. The objective is to have the easiest and most important 75% of the system functionality in the first 90 days, and the rest in subsequent 90-day chunks **[LAWS, 1996]**.

3.4.5.1 The history of RAD

In the mid 1980's, Scott Shultz, a project manager at Dupont, created a new, successful SDM that he called Rapid Iterative Productive Prototyping (RIPP). The idea was to deliver usable pieces of the final system every three to four months, in order of business priority. Eventually these pieces would be used to build up the final system. This iterative SDM permitted on going feedback from users over the course of the system development rather than waiting until the system was complete. James Martin coined the acronym RAD and popularised the approach in his book Rapid Application Development in 1991 **[MARTIN, 1991]**.

According to the Dynamic Systems Development Method (DSDM), RAD, as an SDM grew in a very unstructured way, there was a commonly agreed definition of the RAD SDM but many different vendors and consultants came up with their own interpretation and approach. By 1993 there was momentum in the market place with a growing number of vendors developing or repositioning their products to meet a growing demand from their customers for the RAD SDM. However a piece of the jigsaw was still missing, for every customer that needed RAD tools to improve their development capability, there was a customer who needed to change the development process. It was out of this recognition in the market place for an Industry Standard RAD SDM that the DSDM Consortium was born **[DSDM, 2000]**.

3.4.5.2 The Development of the DSDM Framework

In January 1994, the sixteen founder members of the DSDM met for the first time to discuss the development of a public domain RAD SDM. A framework proposal was made and in March 1994 with consortium strength of 36 members, the framework was approved. The basic concepts agreed in 1994 have remained in place, but the framework has been developed and refined over the life of the consortium. Version 1 of the framework was completed in January 1995 and was published on 22nd February 1995. Together with the publication of the SDM, the consortium had put in place a training scheme with half a dozen accredited training organisations in order for DSDM practitioners to gain certification. In the UK the British Computer Society (BCS) now approves the accreditation and examination procedures and issues the certificates to those practitioners who gain certification.

Once Version 1 of the framework had been accepted and released, the DSDM launched an Early Adopters Programme where users of the RAD SDM could provide the DSDM with feedback on the new framework. This programme lead to the creation of Version 2, which was published on 5th December 1995. The Version 2 framework remained static until January 1997, it was then identified that DSDM was increasingly being used in business process projects and less involved in business application projects. This was felt to be a major shift that should be reflected in the SDM and a Task Group was et up to consider the implications. The outcome from Task Group discussions lead to Version 3 being released on 18th October 1997.

There are no plans for another version in the foreseeable future. Instead the DSDM method will be expanded through White Papers and other publications.

The DSDM Consortium intended to produce the de facto approach to building systems both quickly and well. In the UK, DSDM is claimed to be the most commonly used framework for RAD projects since it has repeatedly demonstrated success in organisations of all sizes from both public and private sectors. It is now gaining acceptance on a worldwide basis.

3.4.5.3 The DSDM Lifecycle

The lifecycle that DSDM uses is iterative. It moves away from the waterfall approach because new technologies have enabled visualisation of the interim products of system development. RAD, through the application of the DSDM framework is best suited to projects that can demonstrate functionality through visual means i.e. screens, reports, etc. According to the DSDM this enables prototyping to be used to maximum benefit. The end users of the project should be easily identified such that knowledgeable representatives can be chosen to participate throughout the life of the project. DSDM calls these representatives "Ambassador Users" as they operate in much the same way as ambassadors by providing a two-way communication channel between the business community and the IT community. A further criterion for project suitability is the ability for large projects to be broken down into smaller sub sections for incremental delivery or for development by parallel teams. Figure 3-16 below illustrates a typical RAD SDM within the DSDM framework **[DSDM, 2000].**

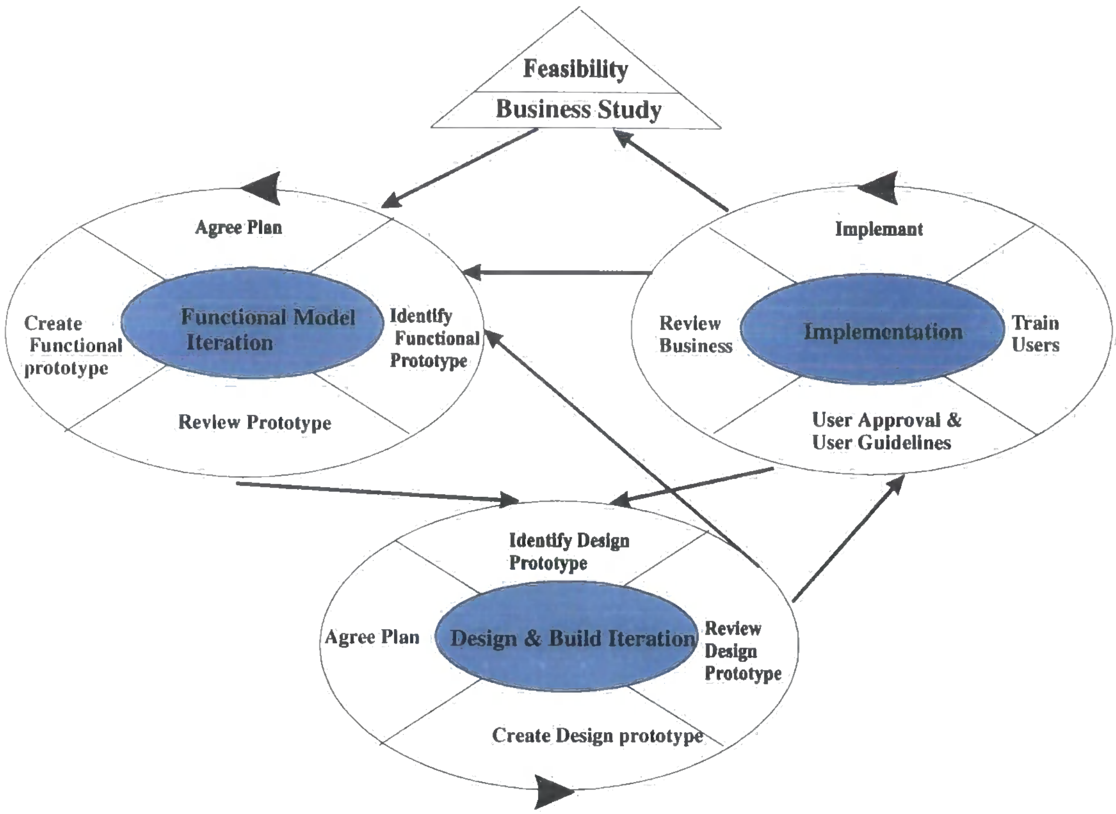


Figure 3-16 - The DSDM Lifecycle [DSDM, 2000]

As with any RAD SDM, the DSDM framework provides a generic process, which must be tailored for use in a particular organisation dependant on the technical, and business constraints.

As DSDM state the above framework should not be used in a prescriptive manner, it may not be suitable for certain organisations. Graham in his study of requirements engineering **[GRAHAM, 1998]**, states that the above model is not entirely suitable for those organisations wishing to adopt an object-orientated (OO) approach to software development. He states that a key characteristic of OO SDMs is the highly flexible nature of requirements gathering, if used within an OO environment Graham states that the DSDM framework can enforce rigidity. Graham goes on to state that the DSDM framework has little or no reuse management, a characteristic that is crucial within OO development ensuring that a system can be added to without disturbing existing components. He states that the prototype stages of RAD would greatly benefit from the use of existing components, helping to reduce developer effort and project timescales **[GRAHAM, 1998]**.

According to Graham, the DSDM framework is very product orientated whereas OO frameworks tend to be process orientated. The use of timeboxing within DSDM means that milestones are equated with deliverables and hence the framework becomes very product based. A further identification made by Graham is the focus that DSDM places on the separation of functional and design iterations. He states that when products are prototyped through the DSDM framework they pass through a fourfold categorisation, 1) business prototypes (or demonstrators), 2) capability/design prototypes, 3) performance and capacity prototypes and 4) usability prototypes. Graham states that this separation of functional and design iterations is incompatible with the more seamless OO approach. OO examines both user interface and overall system design at the same time, thus providing a high level view of how the final product interacts with existing systems **[GRAHAM, 1998]**.

Figure 3-17 below will describe the stages of the DSDM framework

SDM Stage	**Description**
Feasibility Phase	• Examines the suitability of the project for a RAD approach and checks that certain technical and managerial conditions are met. The feasibility study typically lasts a matter of weeks (rather than months).
Business Study	• Scopes the overall activity of the project and provides sound business and technical basis for all future work. • Best carried out as part of a workshop and can usually be done within one month when the right users are involved **[MARTIN, 1991]**. • Within this phase the high level functional and non-functional requirements are base lined, a high-level model of the business functionality and information requirements is produced, the system architecture is outlined and the maintainability objectives are agreed.

SDM Stage	Description
Functional Model Iteration	• This is the main prototyping phase within the framework. • Each iteration of prototype may be navigated a number of times, usually no more than three for each area of the system. • The completed functional model will consist of all necessary high level analysis models and documentation supported by functional prototypes which in turn address detailed process and usability.
Design and Build Iteration	• It is within this phase that each prototype is sufficiently well engineered for use in its operational environment. • A precondition for moving from the functional model iteration into the design and build iteration is an agreement on a part of the functional model. • Some components of a system may well pass from the functional model iteration into the design and build iteration while other components are still very sketchy or even non-existent. • In large RAD projects the implementation may be phased, so system design may be concurrent to some implementation.
Implementation	• It is within this phase that the latest increment is introduced into the operational environment and users of the system are trained. • Differs from waterfall approaches where specifications are written, followed by a detailed design followed by the coding stage and then testing. A typical RAD approach involves the design of an on-screen prototype from which code can be generated directly from the design. • Differs from waterfall SDMs by using the same team for design and implementation. • Involves a review of what has been achieved and highlights any rework if necessary.

Figure 3-17 – Stages of the DSDM SDM [DSDM, 2000]

3.4.5.4 The Underlying Principles of DSDM

The following principles emerged during the initial work of the DSDM Consortium and have been refined and rationalised as a result of the practical application of DSDM. They are the foundations on which DSDM has been based and each one of the principles is applied as appropriate in the various stages of the SDM described in figure 3-17 above **[DSDM, 2000]**.

1. ***Active User Involvement is imperative.*** - DSDM is a user centred approach. If users are not involved throughout the development process delay will occur as decisions are made.

2. ***DSDM Teams must be empowered to make decisions.*** – DSDM teams consist of both developers and users. They must be able to make decisions as requirements are refined and possibly changed.

3. ***The focus is on frequent delivery of products.*** – A product-based approach is more flexible than an activity-based one. The work of a DSDM team is concentrated on products that can be delivered in an agreed period of time.

4. ***Fitness of business purpose is essential criterion for acceptance of deliverables.*** – The focus of the DSDM is on delivering the business functionality at the required time. The system can be more rigorously engineered later if necessary.

5. ***Iterative and incremental development is necessary to converge on an accurate business solution.*** – DSDM allows systems to evolve incrementally; therefore developers can make full use of feedback from the users.

6. ***All changes during development are reversible.*** – To control the evolution of all products everything must be in a known state at all times.

7. ***Requirements are base lined at a high level.*** – Base lining high-level requirements means 'freezing' and agreeing the purpose and scope of the system. This assumes that in the case of frozen high-level requirements no more detail will be obtained by breaking the requirement down further.

8. ***Testing is integrated throughout the lifecycle.*** – Testing is not treated as a separate activity. As the system is developed incrementally, it is also tested and reviewed by both developers and users.

9. ***A collaborative and co-operative approach between all stakeholders is essential.*** – The nature of DSDM projects means that low-level requirements are not necessarily fixed when developers are originally approached to carry out the work.

3.4.5.5 Comparison of RAD and Other SDMs

According to the DSDM consortium **[DSDM, 2000]**, waterfall approaches fix *requirements* (and deliver software which satisfies all them) while allowing time and resources to vary during development. With RAD SDMs the opposite is true, *time* is fixed for the life of the project and *resources* are fixed as far as possible. Figure 3-18 below illustrates this. This helps to ensure that the low-level requirements that will be satisfied are allowed to change. Hence an important product of the business study is a clear prioritisation of the high-level functional and non-functional requirements. More detailed requirements are collected during the latter stages of development these are also prioritised.

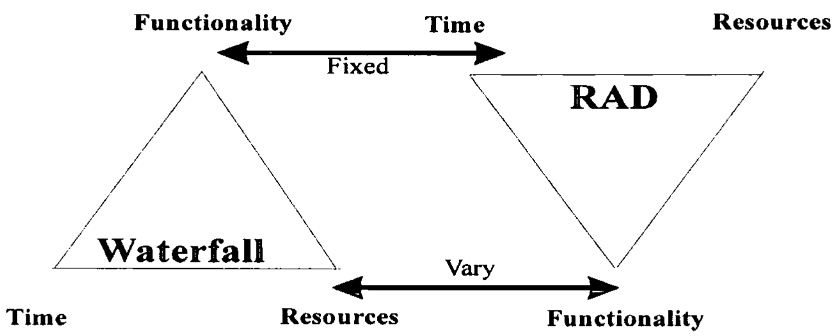


Figure 3-18 – RAD versus Waterfall SDMs

In comparison to the researched SDMs the following observations can be made. The main similarity between RAD and other SDMs is its iterative nature. In this respect it can be compared against other SDMs such as SSM. Both SSM and RAD use an iterative approach to software development. However SSM seems to concentrate more on user viewpoints and less on technical feasibility. In SSM, users are mainly involved in the initial stages of design. Their input is used as a basis for development; a product is then developed and tested against user requirements. RAD concentrates on both user views and technical feasibility by involving users throughout the whole design process including prototyping. The prototyping stage of RAD provides developers with a means of assessing the feasibility of a proposed system with the users before developing a final product.

RAD also shares the same characteristics Boehm's Spiral Model [see section 3.3]. Both the Spiral Model and the RAD SDM do not attempt to start with a full specification unlike the waterfall SDMs. Instead, development begins by specifying and implementing just part of the software, which can then be reviewed in order to identify further requirements. In both SDMs this process is then repeated, producing a new version of the software for each cycle of the model.

In comparison to Rapid Prototyping RAD does provide a means of assessing technical feasibility through some form of iterative prototype. However rapid Prototyping is heavily focussed on assessing the technical feasibility of a prototype whilst RAD examines both user requirements and technical feasibility. Typically in most Rapid Prototyping exercises, users are involved very little with the design process and are more involved in the testing phases of the SDM.

3.4.5.6 Benefits of Rapid Application Development

As illustrated above, RAD was born from failings with previous waterfall based SDMs. RAD aims to provide workable solutions at high speed, high quality and lower cost. This section reports on the claimed benefits of using a RAD SDM.

3.4.5.6.1 High Quality Solutions

So how does RAD provide a higher quality solution? RAD requires several factors in order to improve the quality of the delivered system these are:

- Continuous involvement of the end user at the early stages of the design process **[UCD, 1997]**.
- Prototyping, which helps the users visualise and make adjustments to the system **[MARTIN, 1991]**.
- Use of CASE (Computer Aided Systems Engineering) tools to enforce technical integrity **[MARTIN, 1991]**.

- Use of integrated CASE tools to provide bug free code **[MARTIN, 1991]**.
- User involvement in the Construction phase, allowing prototype to be adjusted if necessary **[DSDM, 2000]**.
- The ability to redefine low-level requirements **[STAPLETON, 1995]**.

One of the main 'quality' factors is that through the involvement of end users RAD helps to identify early detection of issues. Work carried out by DSDM illustrates that RAD is a user-centred approach. If users are not closely involved throughout the development process, delays will occur as decisions are made and users may feel that the final solution is imposed by developers **[DSDM, 2000]**.

In her paper, "A Quality Approach to Rapid Application Development" **[STAPLETON, 1995]**, Jennifer Stapleton states that the general assumption is that business requirements will probably change as understanding increases. Since RAD allows the user to revisit low-level definitions, development time isn't wasted on perfecting initial requirements. As illustrated in principle 7 of the DSDM principle set, high-level requirements should be based-lined and remain as static as possible in order to control scope. From a project management perspective the Butler Group Consultancy **[BUTLERGROUP, 1998]**, state that due to the dynamic nature of RAD, timeboxing is an essential project management aspect to be considered. Timeboxing is discussed in more detail in section 3.3.6.7.5.

The DSDM consortium supports Stapleton's comments and quotes that the focus of RAD is on delivering the necessary functionality at the required time. With waterfall SDMs, the focus has been placed on satisfying the contents of a requirements document and conforming to previous deliverables. Sometimes these requirements are inaccurate, the previous deliverables may be flawed and the business needs may have changed since the start of the project. Hence DSDM state that one of the major underlying principles within any RAD project should be the fitness for business purpose as this is an essential criterion for acceptance of deliverables **[DSDM, 2000]**.

3.4.5.6.2 INCREASED DEVELOPMENT SPEED

As illustrated above, one of the underlying principles of the DSDM SDM is that the focus of any RAD project should be on the frequent delivery of products. The work of a RAD team is concentrated on the products that can be delivered in an agreed period of time. This enables the team to select the best approach to achieving the deliverables required in the time available. By keeping each phase of the project short, the team can easily decide which activities are necessary and sufficient to achieve the right deliverables **[DSDM, 2000]**.

An example of fast development times through RAD is provided by the Fujitsu case study. The Slough-based manufacturing company repairs disk drives. The dramatic rise in the sale of three-and-a-half inch disk products since 1993, and the Japanese manufacturer's increasing success in the disk market, meant that the centre's systems had become too slow and inflexible to cope with the likely volumes of returned units. A new system was required within a very tight deadline.

The General Manager Steve Clarkson realised that the speed of implementation would be vital to the success of the new system and suggested that some form of RAD SDM be used for development. For the purposes of the project Fujitsu partnered with the MDA Computing software house (a member of the DSDM) in order to apply the RAD approach. Mr Clarkson stated: "We needed immediate results to solve pressing problems we could not bear the thought of wasting six months talking before we could see any action. RAD is an extremely quick and attractive way to get a system that works. Other functionality can always be added on later." **[DSDM Fujitsu Case Study, 2000]**.

MDA began its close co-operation with the Slough staff in November 1994, using a tried and tested RAD SDM to ensure the users got exactly what they wanted. Fujitsu European Repair Centre was able to use its new system as early as the following February, although May was the official release date. Some three months after initial project conception, RAD had helped to deliver a working system.

The Post Office provides a further Case study to support faster development times using the RAD SDM. Post Office IT (POiT) supplies technology solutions to The Post Office and interlinked businesses – and as such is one of the largest IT suppliers in the UK. POiT investigated ways of improving productivity and understood that RAD could help them achieve their goals.

POiT was given a rigorous set of targets. These included doubling productivity against waterfall SDMs, and developing the new system to a very tight development budget and very tight timescales. The entire project took seven-and-a-half months. Based on previous projects, POiT commented that a similar project carried out using a waterfall SDM would have taken thirteen months to develop. Thus the reduction in time to market halved the labour costs of the project **[DSDM POiT Case Study, 2000]**.

3.4.5.6.3 INCREASED USER SUPPORT

A key term often used when talking about RAD is that of 'empowerment'. A dictionary definition of empowerment reads:

To empower - ***to authorise or licence, to give power to.*** **[FOWLER & FOWLER, 1990]**

One of the perceived benefits of the RAD approach is the ability to empower users to be an active part of a project's success. According to the DSDM RAD identifies the need to involve a broad spectrum of personnel at different time throughout a project. The DSDM overview states that some users are expected to become an integral part of the development team in order to make day-to-day decisions.

"Users no longer sit outside the development team acting as suppliers of information and reviewers of results."

[DSDM, 1999]

This integration of business and IT roles is fundamental to RAD based projects. Compared to waterfall SDMs, the RAD SDM places a much greater emphasis on the involvement of the end user.

In the Fujitsu case study provided above, Steve Clarkson states – "RAD, by its nature, actively involves everyone, so the system the users get is the very one that they have evolved. It is a big benefit to give people the tool that they have requested rather than foist something on them. Their reaction towards the delivered system and co-operation are totally different. Everybody is for it, and supports it wholeheartedly when the system is delivered." **[DSDM Fujitsu Case Study, 2000]**.

3.4.5.6.4 FREQUENT DELIVERY OF END PRODUCT THROUGH PROTOTYPES

A further documented benefit of RAD is its ability to produce a fast solution through the use of a prototype. It is important to note however that prototyping is not exclusive to the RAD; other waterfall SDMs such as SSADM also use the prototyping technique. However the nature in which prototyping is carried out iteratively within RAD helps to provide early and frequent demonstration of the final product. The use of prototyping within RAD aims to address some of the well-reported problems with traditional software design. Boar highlights this in his example that 20 to 40 percent of all systems problems can be traced to problems in the design process, while 60 to 80 percent can be traced to inaccurate requirements definitions **[BOAR, 1984]**.

3.4.5.7 CONSIDERATIONS TO BE AWARE WITH WHEN USING THE RAD SDM

This section will examine a few of the possible issues that require consideration when using the RAD SDM.

3.4.5.7.1 FINDING THE RIGHT SKILL SET

When any new tool is introduced into an environment there is usually a divide between employees who are immediately enthusiastic about a change who want to jump in and use fresh ideas, there are also those who resist change and would prefer to delay its implementation. When implementing change it is useful for managers to categorise these employees into groups. Martin suggests that the following categories of behaviour should be examined:

Behaviour	Characteristics
Experimenters	Like to try anything new but tend not to stick to using a new method
Early Adapters	Can visualise ways in which a new SDM will help them and tailor the method so that it works best within their environment.
Pragmatists	Middle of the road type people who are not too anxious to experiment, are cautious of failure and don't like to spend too much time learning new SDMs. Will only accept a new method once early adapters have demonstrated that the SDM works.
Late Adapters	Are reluctant to change, people who do not want to learn new SDMs.
Resisters	Forcefully oppose new SDMs and voice their opposition early within a project. Late adapters and resisters tend to be older people within the organisation, who are dignified, influential and also quite powerful.

Figure 3-19 – Behaviours within organisations [MARTIN, 1991]

Martin states that the following percentages of behaviours can be found in a 'typical' organisation: 5% Experimenters, 15% Early Adapters, 35% Pragmatists, 35% Late Adapters, 15% Resisters.

These percentage figures for personality mix illustrate that most IS organisations have a wide combination of employees who posses these characteristics, each of which need to be managed correctly in order for successful implementation of the RAD methodology. In their research of the RAD SDM the ButlerGroup **[BUTLERGROUP, 1998]** state the RAD process requires skilled individuals to play one or more roles and requires increased knowledge of both business and technology. A typical RAD project may require people to perform roles such as facilitator, applications architect, GUI designers/developers, application developers and QA testers. Not all members can know all the tools, methodologies, standards, libraries and technologies, which is why the definition of roles is crucial to the success of RAD.

RAD relies heavily on a skilled team, appropriate training and expertise are as important as the tools themselves. An experienced team member can make the RAD process much more effective. The RAD SDM also relies heavily on the involvement of end users, commitments in these areas are crucial to the effectiveness of the RAD approach **[CHASAN, 1999]**. Often this people focus is overlooked as an invaluable skill and employees are assigned tasks that they are not trained to complete.

The ButlerGroup state that managing user involvement in terms of time, deliverables and commitment becomes increasingly important when dealing with RAD projects. Clear objectives must be set and teams must be empowered to achieve these objectives in their own way. Objectives however must be measurable, for example, if the objective is the prototype it will only have been met if the prototype is accepted by the users **[BUTLERGROUP, 1998]**.

3.4.5.7.2 A Change Of Culture

RAD brings with it a whole new concept of working and it is essential that it is combined within a well-defined lifecycle. When employing RAD it is essential to have a top IS executive totally committed to the project, in some cases this can be a very demanding job with heavy time constraints. For these reasons many organisations can't afford to involve a committed IS executive and hence this requirement is sometimes overlooked. Without support from an IS executive, the RAD approach can prove to be difficult to implement. A further knock on effect of introducing RAD into an organisation is that of 'pay back'. Some of the re-tooling required provides no short-term improvement in profit; its results are long term. This needs to be clearly understood before deciding to implement the RAD approach **[DSDM, 2000]**.

The introduction of computerised systems and new technology into an organisation can bring with it many changes. The implementation of computerised information systems can change the way both managers and organisations perform **[FREDERICO, 1985]**. Given modern managerial methods and ordinary organisational occupations, it is reasonable to expect that the introduction of computerised information systems will have a noticeable effect on managerial performance and decision-making, as well as organisational structure and processes. It is important to understand that the introduction of new systems doesn't just imply new technology; it is beneficial to also examine the wider environmental issues too.

3.4.5.7.3 INTEGRATION

According to Martin, one of the most overlooked aspects of implementing a RAD approach is that of integration, ensuring that RAD fits into the current method of working. He states that usually organisations attempt to use RAD to build stand-alone systems – to solve a business problem in isolation. However in reality most business applications rely on existing databases to support them, most applications consist of a network of interlinked databases. A common infrastructure is vital to the application of RAD as without this consideration reusable code and designs are almost impossible **[MARTIN,1991]**.

Martin continues to state that it is this integration of information resources that enables fast delivery of information and thus, rapid decision-making. DuPont applies the following strategy:

"A fundamental objective of information technology architecture is to make integration as easy as possible, so that time-based competitive advantages can be realised."

3.4.5.7.4 Choosing the Right Tools

A further possible problem to be aware of is the heavy reliance RAD places on the chosen toolset. The DSDM states that it is highly important to find a toolset that does what is required at the onset of a project. It is increasingly difficult to switch between toolsets half way through a project, not only are new techniques required but existing work may not be compatible with the new tools. The choice of tools required should be made very early on in the project lifecycle before people have climbed the learning curve with an inadequate toolset.

Chasan states in her study of RAD that many IS managers often make the shortsighted mistake of economising by limiting software tool purchases. As with any purchase, it is often better to spend money up front to get a better tool than it is swap tools half way through a process when it is realised that the tool is inadequate. Chasan states that an effective tool can mean the difference between a dynamic, responsive RAD team and a struggling, stressed, ineffective team **[CHASAN, 1999]**.

3.4.5.7.5 Project Management

As illustrated previously by the RAD lifecycle, the core of the system is built quickly and then refinements are added to it. A system will be built up in a number of layers until it meets the users' requirements. This onion-like refinement makes the setting of deadlines highly crucial within a RAD environment. For this reason DSDM suggests that some form of 'timeboxing' must be applied in order to set tight deadlines within which the system must be delivered. The concept of timeboxing will now be explained.

In any project, there is a fixed completion date, which provides an overall timebox for the work to be carried out. DSDM refines the concept of timeboxing by nesting shorter timeboxes of two to six weeks within the overall time frame. It is these nested timeboxes that are the focus for monitoring and control activities **[DSDM, 2000]**. If the project starts to slip the emphasis in RAD projects is on reducing the requirements to fit the timebox, not increasing the deadline. Figure 3-20 below illustrates the timebox concept.

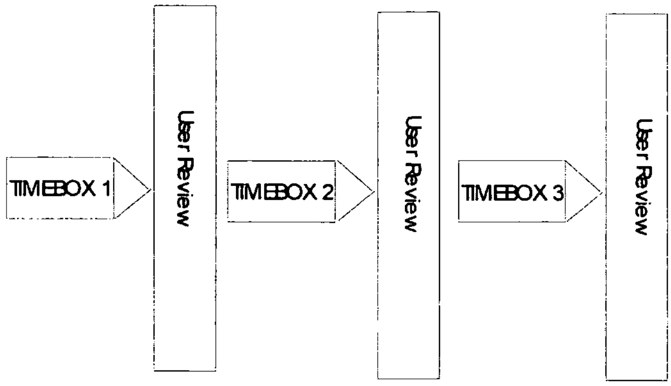


Figure 3-20 – Timeboxes and User Reviews [DAVIES, 1998].

3.5 Management Information Systems (MIS)

In order to fully understand the need for the Philips MIS this section will provide an explanation of the objectives of an MIS, what they are and how they are used. Although the Philips MIS is classed as a Manufacturing Information System, the theory behind the project is based upon Management Information Systems.

An IS can be defined as a set of interrelated components that collect (or retrieve), process, store and distribute information to support decision making and control within an organisation **[LAUDON & LAUDON, 1999]**. Figure 3-21 illustrates Laudons perception of an Information System. Many definitions and theories exist however a common element of 'networked components' is always present. Murdick and Munson provide a second definition of an MIS.

"A system that monitors and retrieves data from the environment, captures data from transactions and operations within the firm, filters, organises and selects data and presents them as information to managers."

[MURDICK & MUNSON, 1986].

Murdick and Munson describe the main purpose of an MIS to be to raise the process of managing from the level of intuitive guesswork and isolated problem solving to a level where data can be processed in a more sophisticated manner. They continue by stating that managers have always had numerous 'sources' of information; the MIS provides a system of information to allow managers to use these sources to make decisions and solve problems.

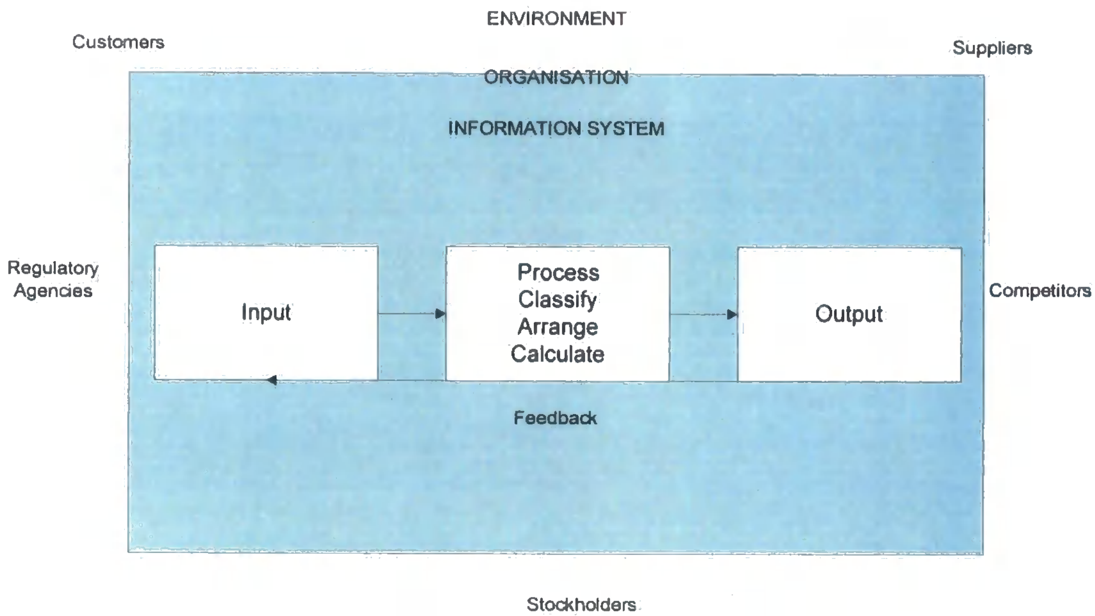


Figure 3-21– Components of an Information System [LAUDON & LAUDON, 1997].

Three activities in an IS produce information that organisations need for making decisions, controlling operations, analysing problems and creating new products or services. These activities are input, processing and output. The **input** stage of figure 3-21 is where raw data is captured from within the organisation or its environment. **Processing** converts the raw data into a more meaningful form. The **output** stage transfers the meaningful information to the people or activities where it will be used. All information systems require some form of feedback loop in order to enable the monitoring and evaluation of the output data.

An IS can be either computer based or manual. Computer based information systems rely on computer hardware and software technology to process and disseminate information. Manual systems tend to use paper and pencil technology. The Philips MIS consists of both types of information system.

3.5.1 A Business Perspective of Information Systems

Information systems should not be studied in isolation; the environment within which the IS operates must also be examined. From a business perspective, an information system is both an organisational and management tool. This management tool is based upon information technology and it is used to combat the challenges forced by the external business environment **[LAUDON & LAUDON, 1999].** In order to understand information systems a business manager must not only understand the latest technology but also the wider organisation, management and information technology dimensions. Figure 3-22 below illustrates the business perspective.

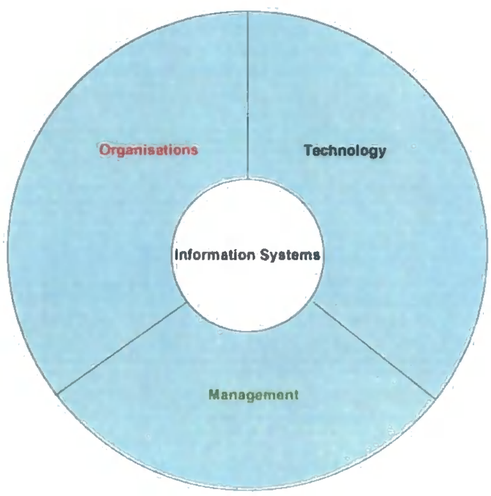


Figure 3-22 – A Business Perspective on information Systems [LAUDON & LAUDON, 1999]

3.5.2 External Impacts on IS

As can be seen from figure 3-22, IS should not be treated in isolation; one important aspect to consider is that of technology. As advances in computer technology increased so did the requirements of information management. Enterprise Relationship Planning or ERP is a good example of this within the manufacturing industry. In the mid to late 1990's, especially within manufacturing industries the purchase of materials became the driving force for profitability **[PTAK & SCHRAGENHEIM, 2000]**. Integrated resource management became the focus for a competitive company and an automated materials management method known as

Materials Requirement Planning or MRPII was born. Computer technology promised automation and manual planning soon became a thing of the past. Through MRPII, inventories were made visible to anyone with a computer and for the first time the computer could calculate what materials an organization required in order to complete a task. No longer was it tolerable to submit a request to the IT department and wait 9 man months of programming time to get critical information. Methods of information management needed to change, information was now required in shorter time frames in order to make good business decisions.

The cost of technology started to fall rapidly and the advent of the PC revolutionized the face of business management systems. The power of these small PCs far exceeded that of the large mainframes and client-server technology quickly replaced older systems. It now became possible to run a fully integrated MRPII system on a small PC. This allowed not only small companies to utilize this new approach to computing but the larger companies then began to move quickly from centralized mainframe systems to agile client-server systems. This move to more client-server focused technology lead to a whole new way of thinking about computing in general, especially resource planning and in the late 1990's the next stage in resource management; an application known as ERP or Enterprise Resource Planning evolved. It is important to note however that ERP is not purely a planning tool. It was designed to include resource planning for a company, including product design, information warehousing, material planning, capacity planning and communication systems. These critical business issues affect non-manufacturing industries too, any company that wishes to achieve competitiveness by best utilizing their assets would benefit from applying ERP.

In their study of ERP, Ptak and Schragenheim **[PTAK & SCHRAGENHEIM, 2000]** state as the power and sophistication of the computer continues to grow, the continued development of tools and techniques to collect data, provide information, and to better manage the enterprise can be expected. Buck-Emden in his study of ERP **[BUCK-EMDEN, 2000]**, states that ERP through modern information management, helps to minimize costs, improves quality and provides crucial competitive advantages.

3.6 SUMMARY

This chapter has documented the various SDMs that have been examined throughout the development of the Philips MIS. This chapter has also helped to highlight the need for a more flexible; user focussed SDM due to the heavy 'people based' focus of information systems in general.

The information provided on the RAD SDM aims to highlight the objectives behind the choice of this SDM over more traditional techniques, the application of which will be documented and analysed in Chapters 6 and 7.

CHAPTER 4

THE DEVELOPMENT PROCESS & METHODOLOGY APPLICABILITY MEASUREMENTS

4.1 INTRODUCTION

This chapter is split into two sections. Firstly it aims to provide the reader with an insight into the development process that was used throughout the MIS project. This is intended to highlight the different nature of the various stages of development to which the RAD SDM was applied. Secondly this chapter documents the various measurements by which the RAD SDM was assessed. These measurements have been chosen in order to evaluate the perceived attributes of RAD, documented in Chapter 3.

4.2 THE DEVELOPMENT PROCESS

This section will describe the stages through which the MIS was developed. Due to decisions made prior to the commencement of the project, the MIS was developed using predominantly RAD based techniques. However throughout the project many stages also involved the use of other techniques combined with the RAD SDM. Each development stage and the SDMs used will now be explained. Once the MIS had been implemented RAD was then used to identify possible enhancements to the system. The RAD SDM was also used in the maintenance of the MIS system.

4.2.1 REQUIREMENTS ANALYSIS

Initially the project began with a requirements analysis, a study to determine the key requirements for the system. The methods used within requirements analysis are documented later in section 4.3 (measurements of RAD applicability). Requirements analysis was carried out using a number of differing SDM techniques. Softer, less structured approaches were combined with more structured SDMs such as the Waterfall approach and the more iterative technique of RAD. By using a softer, less structured analysis approach, this provided a method by which to initially assess the 'human' aspect of the required system. The resulting information gained from this approach was then fed into the traditional lifecycle for further analysis and modelling.

Finally, the combination of results gained from previous SDMs were then used as an input to the RAD approach. RAD provided an overall framework assessing both the 'human' and technical aspects of the required system. Rapid prototyping techniques were used initially to assess the feasibility of the project the results of which were then fed into a RAD approach. These two methods helped to produce a prototype of the required system allowing users to see the proposed end system and providing developers with another requirements gathering tool. An example of the prototype can be found in section 4.3.10.

4.2.2 PROTOTYPING

"If you can determine exactly how much something will cost, exactly how long it will take, and exactly what the result will be, you are doing the same old thing again"

[MULLIN, 1990].

As previously discussed, a rapid prototype of the proposed system was designed based on the input from the requirements analysis stage. The prototyping stage utilised Rapid Prototyping through RAD. Once initial requirements had been gathered through the methods mentioned previously, a rapid prototype was designed in MS Access. The prototype was initially developed by IT in close connection with system users. Once implemented, it was then used as a basis for final design, providing a visual means of testing the system and providing feedback for further development. Figure 4-4 illustrates the MIS prototype.

The prototype developed as part of the MIS study evolved through several iterations, the first of which produced a storyboard or throwaway version of the final system. Once developers and end users were happy with the requirements gained from the storyboard exercise, a more RAD approach was taken by developing 'Language based' prototypes.

4.2.3 System Testing

The RAD approach again was dominant within this stage of the software lifecycle. The prototype produced from previous phases was now used as a testing tool to determine the usability of the designed system. System testing was split into two main areas, internal acceptance testing (for developer use) and end user testing. Once developers and end users were happy with the designed system, a series of testing sessions were run to ensure that the system functionality behaved as intended. This internal (functionality) testing helped to ensure that the system could deliver what was promised before end users tested it in the intended environment. End user testing allowed the development team to determine if the system could operate effectively under the exact situations that it was intended for. The testing stage again was iterative in nature allowing the development team to assess the functionality of the system whilst also allowing for corrections and system enhancement.

4.2.4 System Rollout (Implementation)

Once the design had been developed, coded and tested a phased implementation took place across the factory using a RAD approach. It is important to note here that at this stage, all code generation had been completed and 'implementation' refers to installation of the system 'live' onto the factory floor. A phased implementation was favoured over a 'big bang' or 'all at once' method due to the large changes in working practises imposed by the new system for employees. After examination of the imposed cultural changes it was decided that the system be implemented across the largest production line first. This allowed developers to determine if the system could handle the maximum capacity of intended data whilst ensuring the correct procedures were in place for the smooth running of the implementation. Implementation then took place on the remaining lines around the factory over a period of 3 months. Fig 4-1 below illustrates the implementation period for the system.

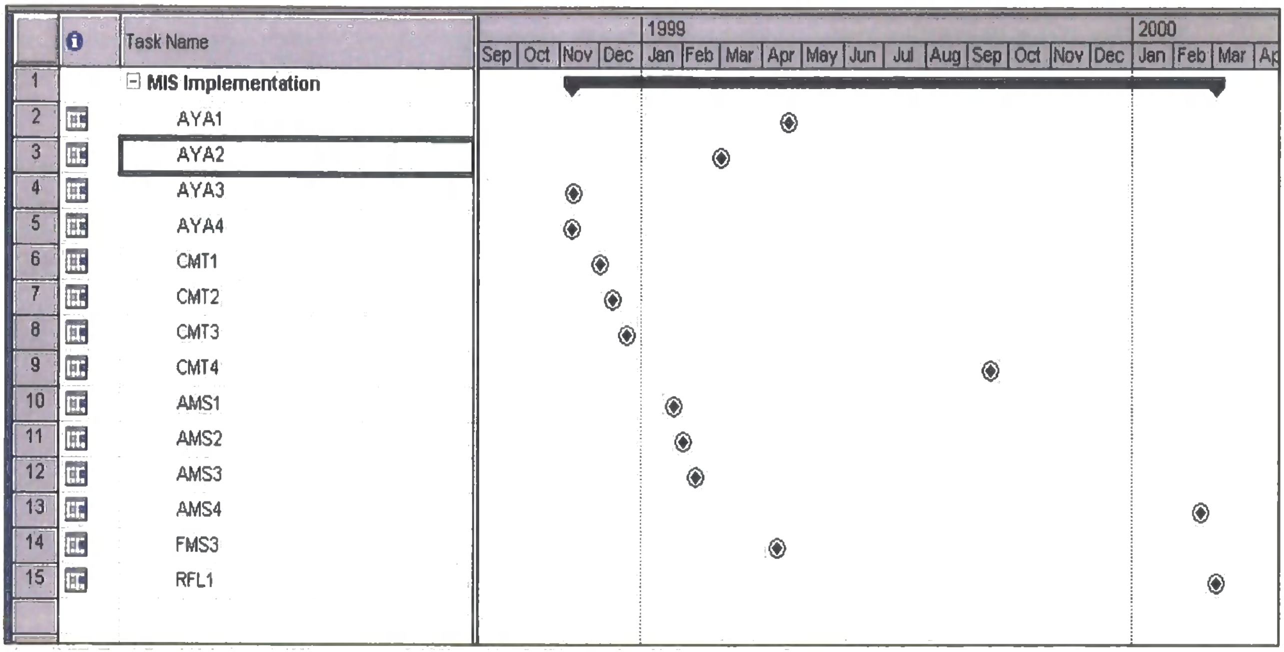


Figure 4-1 – MIS Implementation Period

The initial implementation across larger lines (AYA4,AYA3,FMS3) involved a higher amount of implementation time compared with that of smaller lines (AMS, CMT). As time went on the implementation period decreased for each line, this can be attributed to two things. Firstly any problems experienced during the implementation of earlier lines were identified and solved as soon as possible to avoid duplication of these problems on future implementations. Secondly most of the larger lines with high complexity were targeted for implementation first. Smaller, less complex lines such as CMT and AMS were implemented towards the end.

4.2.5 Summary of Development Process

Throughout the development lifecycle, RAD was the main SDM used. However RAD was seldom used in isolation, in nearly all stages other SDMs were also applied in combination with RAD. This combination of methods helped to provide a more tailored solution for design of the system. In the case of the MIS project however other more suitable SDMs could have been chosen. Chapter 6 analyses the performance of the RAD SDM.

4.3 Measurements of RAD Applicability

This section will document the various measurements that were used to ascertain the applicability of the RAD SDM. Each measurement will be explained with an example and reasoning behind the choice of measurement will be provided.

4.3.1 The Issues Database

Approximately three weeks after MIS implementation it became clear that some method of recording system issues was required. Initially, whenever a problem was encountered users would record issues on a piece of paper and inform developers about their issues the following day. This process tended to be very time consuming and not very effective as issues would get misplaced and never solved.

4.3.1.1 Method

To overcome these problems an issues database was designed for use by the whole factory. Once issues occurred they could be entered directly into a central database. The database is a concurrent tool and is still used at present to record any pending issues. Figure 4-2 illustrates the Issues Database interface.

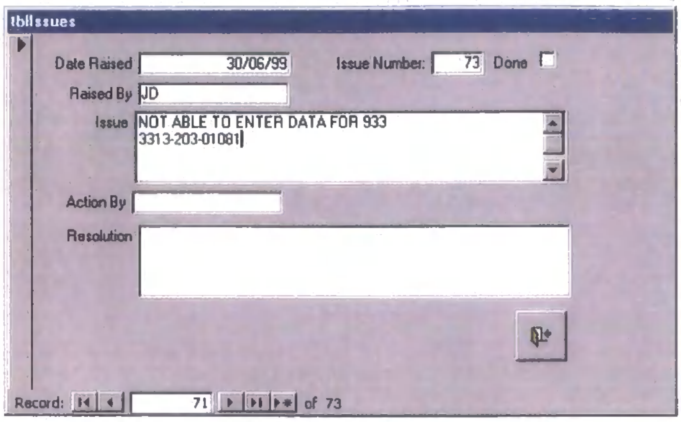


Figure 4-2 – The Issues Database Interface

4.3.1.2 Objectives of The Issues Database

As with most of the measurements applied throughout this study there are two main objectives for the issues database.

- used as a tool throughout development to provide an audit of system issues and to highlight any problems with the system to the developers.
- used as a tool for the purposes of research in order to ascertain the characteristics of the requested changes.

This tool provided the following benefits:

- It allowed IT to determine how RAD handled the types of issues being raised
- It examined how RAD handled the timeliness of issues and the involvement of the user in the construction and design of the system.
- It provided users with visibility of progress regarding existing issues
- It provided IT staff with an audit of issues throughout development

4.3.2 Version Control

For the first three months of development of the MIS project the MS Visual SourceSafe version control tool was used in order to keep track of system changes and to provide rollback in case of development problems. This tool was also used as a measurement to assess project characteristics such as development speed, frequency of system changes and re-engineering techniques. The version control tool also provided a method by which to assess the utilisation of re-usable components within the MIS system.

4.3.2.1 Method

The tool provided a graphical method of managing the project without affecting the base code. Figure 4-3 below provides an illustration of the SourceSafe environment. With reference to Figure 4-3 SourceSafe allowed objects to be 'checked out' and worked on by developers (illustrated by a tick). A lock symbol illustrates objects currently under SourceSafe control. Once an object had been 'checked out' and amended by a developer, SourceSafe would log the name of the developer and request that a reason for the change be entered into a dialogue box. At any stage throughout the project a history of changes could be accessed through SourceSafe.

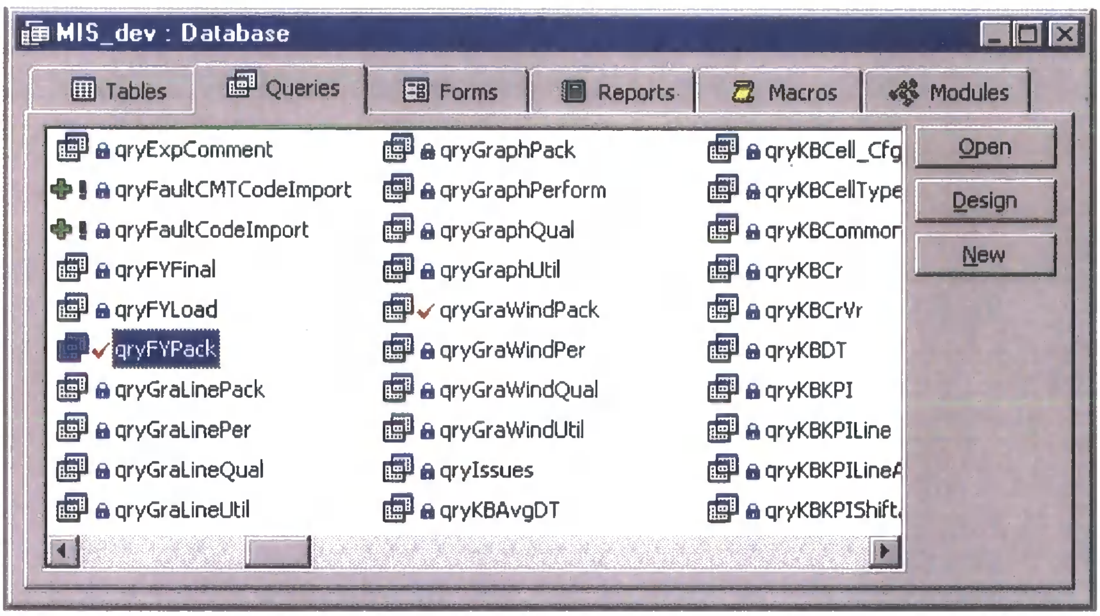


Figure 4-3 – MS Visual SourceSafe

4.3.2.2 Objectives of SourceSafe

There were four main objectives of the SourceSafe tool.

- it was used as a project management tool in order to record and control changes to development.
- it was used as a research tool as part of this study to categorise system amendments and assess frequency of changes to the system.
- It was used as tool by which to assess how RAD affected system development
- It allowed IT to measure the types of system changes that occurred, the stage in the project at which they occurred and the frequency of change.

4.3.3 Interviews/Meetings/User Sessions

Interviews and meeting were held on a regular basis throughout development mainly within the requirements analysis stage.

4.3.3.1 Method

Initially before the MIS project began several meetings were held to identify the main aims of the MIS and who should be the key decision-makers. The RAD SDM aims to ensure that key decision-makers are involved in system design. Throughout the project as milestones were reached and as decisions were required, meetings and interviews were held to ascertain system requirements and issues. The core personnel usually involved in the meetings were: key MIS developers, the project champion (owner) and key customers of system.

Meetings would often take place on a need by need basis and were initiated by the IT department. The content and structure of the meetings was largely situational and dependent on the nature of the topic to be discussed. Any documentation resulting from the meetings was placed on the shared network directory for all users and developers to access.

4.3.3.2 Objectives of Meetings

Meetings were used for both research purposes and throughout the development process. The main objectives were:

- to gain feedback on system use and user requirements.
- to assess one of the key characteristics of the RAD approach, increased user support.
- to measure how RAD could be used to gather initial project concepts, handle feedback and select the right people for the project.

4.3.4 MIS Training

One of the main quoted benefits of the RAD approach is that of the end user support that is provided. It was decided that one measurement of how the RAD approach helped provide this support would be to assess how the end users were trained throughout the project. Training was assessed by an examination of the following aspects: number of employees trained, total training time throughout the project and delivery of MIS training prior to MIS usage.

4.3.4.1 Method

Throughout the MIS implementation a training log was kept to enable the monitoring of training across each line within the factory. Once a training session had been completed, the MIS trainer would update the log with the name of the employee, the shift they were working on and how many hours that they had been trained for. An example of the MIS training log can be found in Appendix B.

4.3.4.2 Objectives of MIS Training

Training sessions were used throughout the MIS project for two reasons.

- used as a development tool to ensure that the MIS users were receiving the required support they needed in order to operate the MIS system.
- used as a research tool to investigate the claims that RAD provides a heavy focus on user support.

4.3.5 MIS User Guide

A further method used to assess how the RAD methodology places a heavy focus on user support was the analysis of supporting documentation for the MIS in particular the MIS user guide. The user guide formed the main supporting documentation for the operation of the MIS and provided the user with a step by step guide on how to use the MIS system. The user guide consisted of a paper based document written in MS Word illustrating visual screen displays of the MIS screens together with a textual description of their operation.

4.3.5.1 Method

Approximately two weeks before the MIS system was implemented across the factory, the user guide was constructed and a copy printed for each production line. Following each training session, a user guide was placed onto the production line in order to provide additional support for the MIS users. An example of a page from the MIS user guide is provided in Appendix D.

4.3.5.2 Objectives of the MIS user guide

As with the majority of the measures documented in this section, the MIS user guide served two main purposes.

- used as a development tool to provide MIS users with additional support in the operation of the MIS
- used as a research tool in order to assess the claims that RAD places a heavy focus on user support.

4.3.6 End User Personality Mix

Due to lack of available resources some of the more 'people focused' studies could not be carried out as part of the overall project development. For this reason some of the techniques listed by Martin in Chapter 3 were applied after MIS implementation. One of these studies included an examination of the different personalities that made up the end user group.

4.3.6.1 Method

The main input to this study was the results provided by the data entry investigation study (see section 4.3.8 for further detail). End users (both managerial and operational) were examined with respect to their feelings on the use of new tools and techniques as part of their working practices. Information was gained on factors such as personal preferences for the use of the MIS and feelings on the introduction of new methods etc. All information gained was then entered into a pie chart based on the model provided by Martin in Chapter 3.

4.3.6.2 Objective

The main objective of this study was:

- to ascertain if any particular user group concerns could have been addressed prior to project initiation and hence assist the end user management aspect of the project. Resultant information from this study can be used as an input to future phases of the MIS project.

4.3.7 User Motivation Study

The user motivation study was another user-focussed technique that was applied after MIS implementation. It was intended as a tool to assess the factors that 'motivated' the end users to use the MIS system and hence target the final system to meet these motivations.

4.3.7.1 METHOD

Users (both operational and managerial) were interviewed with respect to their personal opinions on the MIS system. Once a satisfactory cross section of users had been interviewed the resultant information was then entered into pie charts in order that the key system 'motivators' be examined. In future phases of the MIS project, this information could be used to gain a better understanding of the users' needs.

4.3.7.2 OBJECTIVES

The main objectives of this study were:

- to identify the main motivations for using the MIS. This information could then be used to ensure that the final system met the users needs by satisfying their motivations.

- to identify and discuss any hidden personal agendas that may effect the outcome of the project.

4.3.8 DATA ENTRY INVESTIGATION

In order to ascertain how successful user empowerment through RAD had been and the effectiveness of the MIS training, every month a series of tests were carried out to graphically illustrate the results. A set of MS Access queries were developed, providing results on the following: the percentage of data entry completion across lines over a number of weeks, the time taken to enter the required data across each line and the MIS system accuracy compared against existing systems.

4.3.8.1 METHOD

Once MIS had been implemented across all lines within the factory, tests were carried out over a number of months primarily to determine how well the users were handling the new system. Data volumes from the MIS database were extracted and averaged across each line together with time taken to enter MIS data. The extracted information was then graphed to produce a visual representation of the trends. These graphs were then used at improvement meetings to highlight any areas of concern and provide feedback on how system development was running.

4.3.8.2 Objectives of Data Entry Testing

Data entry testing had the following objectives:

- used as a measurement to assess how effective end user management had been and how the end users were using the system.
- used to assess the 'User Empowerment'/'User Focus' characteristics of the RAD approach whilst also examining how RAD handled the issues of organisational change.
- used throughout the development process to assess the accuracy and timeliness of data being entered into the MIS system.
- provided the development team with a fairly accurate picture of how happy end users were and helped to determine if the system satisfied user requirements.
- used to highlight any areas of concern to Line supervisors.

4.3.9 Change Proposal Forms (CPF)

A CPF was developed in order to provide a method of recording changes to existing MIS objects as well as the introduction of new requirements. An example of a Change Proposal Form can be found in Appendix D.

4.3.9.1 Method

Whenever changes to existing MIS functionality were required, the user proposing the change would arrange a meeting with a member of the IT MIS development team. The requirement would be discussed and its urgency/importance determined. If the request was deemed to be purely cosmetic with no effect on functionality, the change would be agreed and implemented by IT. If the change incurred changes to system functionality it would be recorded on a CPF, signed by the originator and IT and then forwarded to the appropriate department for authorisation. Once authorised and agreed an implementation timescale would be determined and implementation of the change took place. All CPFs were filed for future reference.

4.3.9.2 Objective of the Change Proposal Form

The main objectives of the CPF were:

- to provide an assessment of the frequency and nature of the changes that were requested throughout this RAD based project. This facilitated a measurement of how a RAD approach handled various changes throughout the project lifecycle and assess if the application of RAD provided a better quality product.
- To provide both the development team and end-users an audit of system changes and proposals. Examination of Change Proposal Forms provided an illustration of system changes, timings and frequencies to support activities carried out within the Version Control metric.
- To provide a measure of project control within a RAD environment.

4.3.10 Prototypes

As discussed in section 4.2.2 a prototype was developed as part of the requirements analysis and development stages of the MIS project. Figure 4-4 illustrates the prototype that was developed.

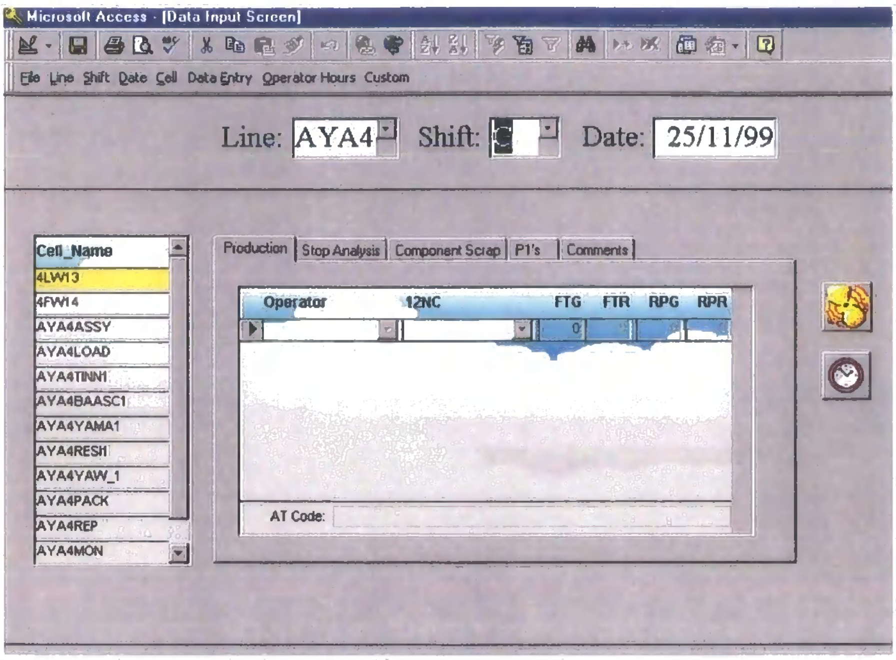


Figure 4-4 – MIS Prototype

4.3.10.1 METHOD

Once the initial project requirements had been gathered, an MS Access prototype was mocked up to illustrate some of the basic functionality that could be provided as part of the MIS system. This prototype however was not made up entirely of visual screen shots, each screen had the required functionality coded into it from the start. This allowed the development team to test the prototype at an early stage within the project. As the project progressed this initial MS prototype was then used as a building block on which to add additional functionality as required. Once each version of the prototype had been tested by IT and the user group, a release was made and that version became operational on the factory floor. Figure 5-28 illustrates the number of prototype releases that were made throughout the project.

4.3.10.2 OBJECTIVES OF THE PROTOTYPE

There were two main reasons for the development of the MIS prototype.

- to test how RAD aims to provide the user with a working model to meet and question user requirements at an early stage within the project
- to provide an iterative tool to obtain requirements as a baseline for the final system.

4.3.11 TCS Mini Project Investigation

As part of the overall study, a mini project was carried out investigating the need for more accurate communication methods and hence the need for an information system.

4.3.11.1 Method

The mini project involved a two-week analysis of the main communication problems facing Philips. Various research methods were used to gain results such as Interviewing, Training and Analysis of existing documentation.

Interviewing employees on a one to one basis was used with an aim to obtaining quick access to directly relevant information. Due to the short nature of some of the interviews this technique limited analysis to small amounts of information. Training users as part of the MIS was used as another analysis tool in order to gain large amounts of specific line information. Finally analysis of current documentation was used with an aim to providing a high level view of the current problems with communication systems in place around the factory.

4.3.11.2 Objectives of Mini Project

The mini project was used as part of the development process to provide grounding on which to base the requirements analysis of the MIS project.

4.3.12 Project Plans

Project plans were used throughout the development of the MIS to assess and manage the progress of the project. MS Project was used to provide a visual representation of the project plan together with information regarding resources, task duration and baselines to assess how the plan had changed.

4.3.12.1 METHOD

Initially a high-level project plan was produced documenting the overall objectives of the MIS project; this plan illustrated the main tasks that required consideration. At the beginning of each project phase, the high-level project plan was broken down into a more detailed representation of each particular phase. This provided the development team with a more controlled method of managing the progress of each project phase. With the initial creation of the project plan a baseline was built into the plan to highlight variations on the plan. Whenever a project plan was updated, a new baseline would be added to plan to illustrate that a change had been made. An example of an MS project plan can be found in Appendix E.

4.3.12.2 OBJECTIVES

The use of MS Project served two main purposes.

- it was used as a project development tool to assist in the planning and management of the MIS project, providing a means by which to assess the progress of the project.
- it was used as a tool to measure the overall impact that the application of RAD had on overall project timescales. Several project plans were produced each of which were used in the various analysis techniques documented in the Results Chapter.

4.4 SUMMARY

This chapter has helped to identify the development process that was followed for the MIS. It documents the stages that were followed and methods and objectives of each. This chapter has also helped to identify the various measurements that were used in order to ascertain the applicability of the RAD approach. Due to the diverse nature of the approach, several measurements were used to examine the effectiveness of the RAD approach within the Philips situation.

CHAPTER 5

RESULTS

5.1 INTRODUCTION

This chapter documents the results and findings observed throughout the course of the study. Analysis and discussion relating to these results can be found in Chapter 6. This chapter illustrates the practical application of the SDMs identified in Chapter 3 and intends to quantify the various methodology characteristics documented. For the majority of the MIS project lifecycle, the RAD SDM was applied, for this reason the results illustrated in this chapter predominantly assess the perceived characteristics of RAD. These characteristics were assessed against the metrics specified in Chapter 3. This chapter is split into the following sections, examining how each method was applied and the associated outcome.

- **TCS Mini project**
- **Examination of Change Requests**
- **Change Management using RAD**
- **Data Entry Investigation**
- **Project Management within RAD**
- **RAD as a prototype tool**

Figure 5-1 below illustrates how each measurement was used within the project lifecycle and the milestones within each measurement. This provides the reader with an understanding of the timescales that each measurement was used for. This diagram also provides the reader with information regarding the timescales of the various SDMs.

For all results provided in this chapter, analysis and further discussions can be found in Chapter 6 Analysis.

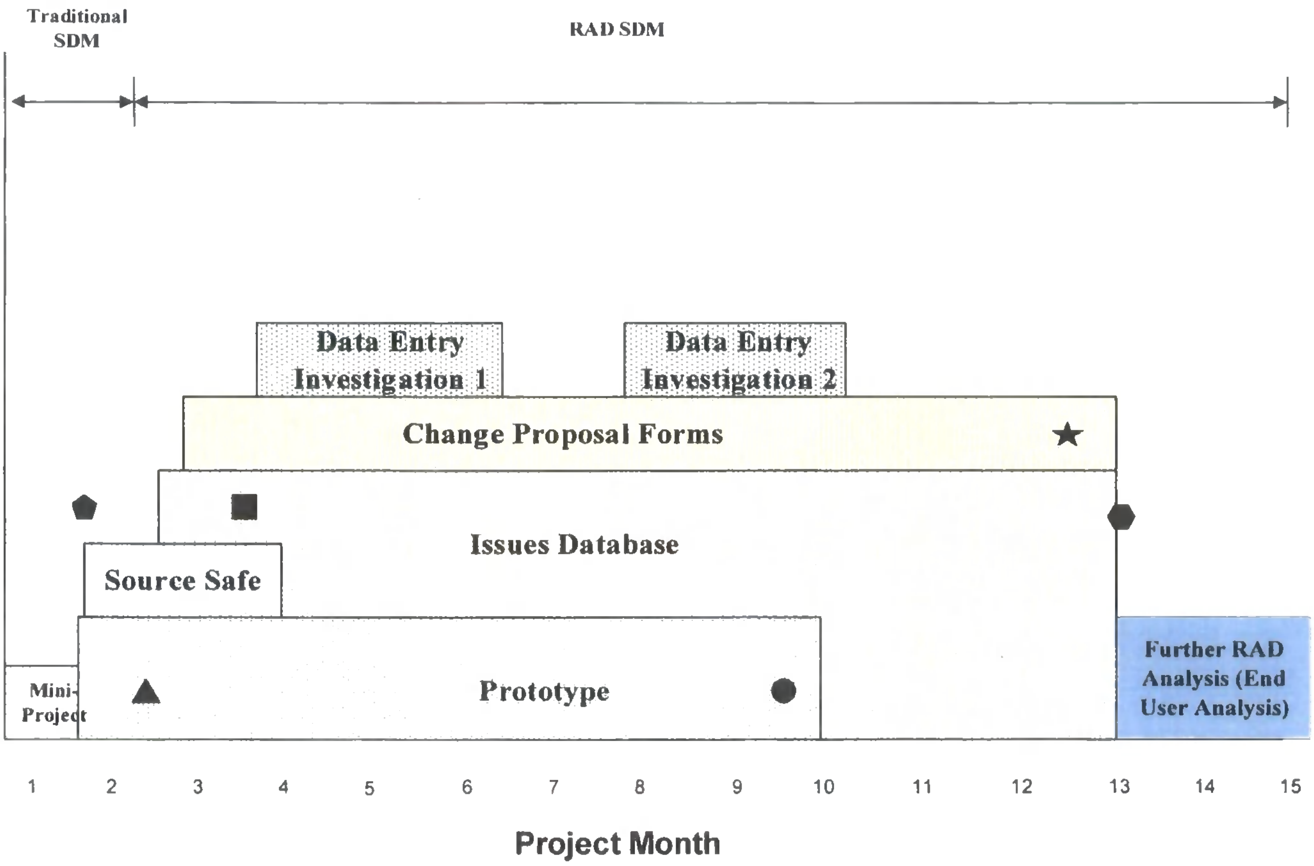


Figure 5-1 – Timeplan of analysis activities.

Timeplan Milestone Key

Initial Prototype Release	Final Prototype Release
Initial Issues Database Release	Final Change Proposal Request
First Change Request	Final MIS Implementation

5.2 TCS Mini Project

As discussed in Chapter 4 a two-week mini project analysis examining the current communication problems within Philips was carried out prior to the MIS project initiation. This section is split into two main areas. Firstly a review of the analysis methods used throughout the study will be provided. Secondly an examination of the key findings from the study will be included. The intention of reference to this 'mini-project' is to provide the reader with an overview of the problem situation prior to the MIS study. The mini-project is also included to provide an example of how the RAD SDM was applied to a very short timescale study.

The mini project involved a two-week RAD based analysis of the main communication problems facing Philips. This study formed part of the overall RA stage within the MIS project.

5.2.1 Analysis Methods Used

Throughout the mini-project several analysis methods were applied. Due to the short timescale of the project, the analysis methods chosen were selected to provide the maximum amount of research within a short space of time. If time had permitted further research methods would have been applied.

Figure 5-2 below illustrates the timescale and analysis methods applied within the Mini Project.

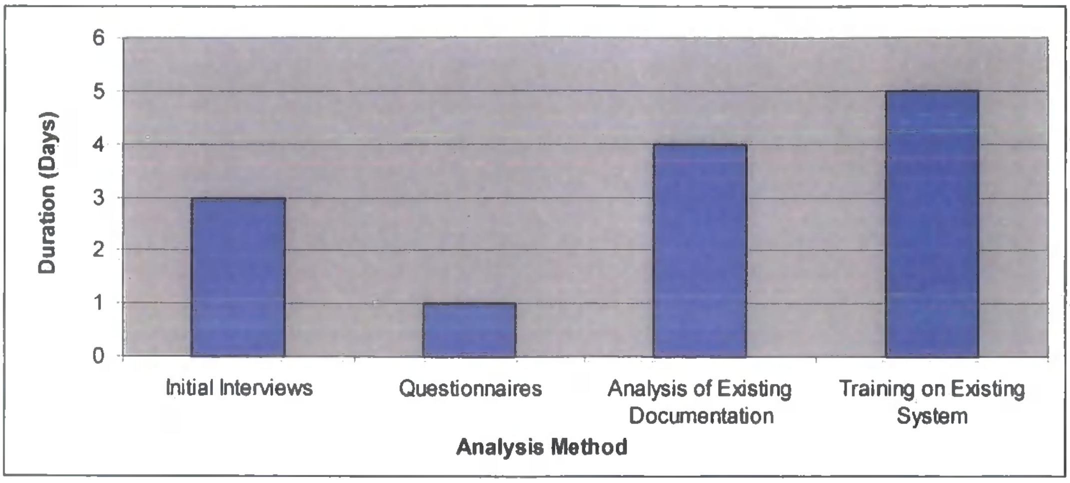


Figure 5-2 – Average analysis time spent on TCS Mini project

SURVEY STATISTICS

The survey of communication within the Philips factory was based upon information gained from interviews, questionnaires, existing documentation and training sessions. The employees involved were selected and screened to ensure that their involvement was appropriate to the survey. Participating employees were drawn from a selection of production concerns, including engineers (10%), line operators (60%) and line managers (30%). Three quarters of the sample had some form of direct involvement with the MIS project.

The investigation of communication methods were split into two main areas, firstly communication from a management perspective was addressed followed by an investigation of communication around the shop floor.

Figures 5-3 and 5-4 below illustrate the 'key areas' of concern that were identified from a management and operational perspective.

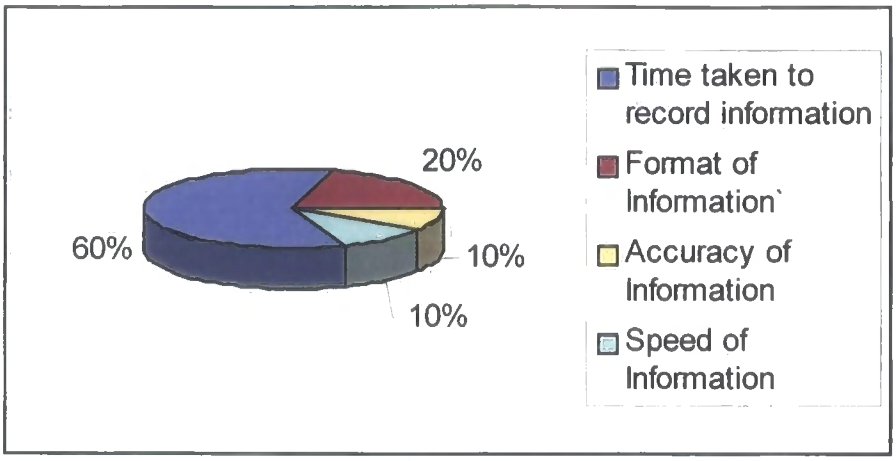


Figure 5-3 - Key areas of concern relating to communication as expressed by management

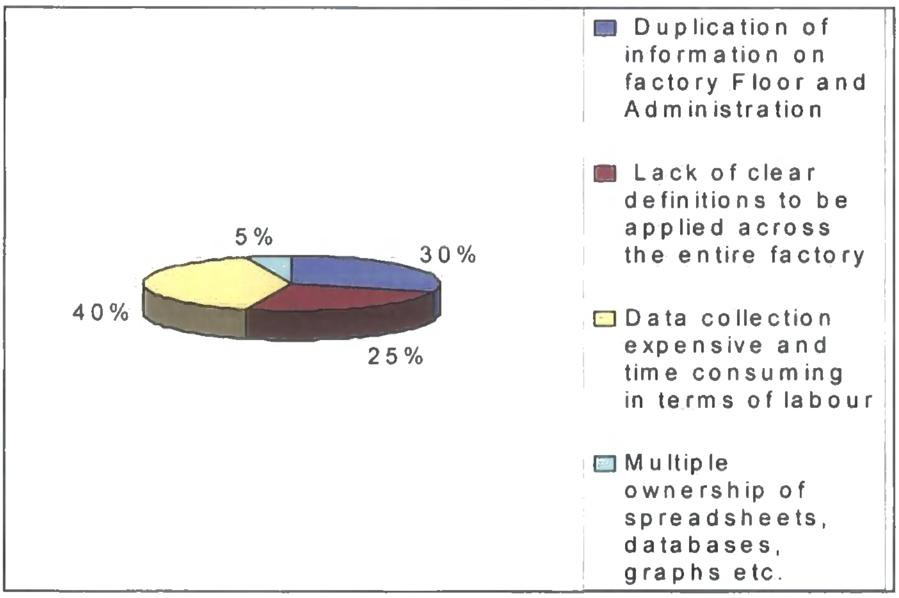


Figure 5-4 - Key areas of concern regarding communication as expressed by operational staff

5.3 EXAMINATION OF CHANGE REQUESTS

The results documented in this section are intended to provide a comparison between the more structured waterfall SDMs and RAD when applied to the assessment of functionality changes throughout the project lifecycle. The main measurement used to assess the applicability of each SDM was the assessment of changes to functionality. This measurement helped to illustrate how each SDM handled frequently changing requirements. This section also examines whether RAD helped to provide a higher quality solution by ensuring user requirements were fully satisfied.

5.3.1 THE WATERFALL SDM

As with most major IT based projects, a starting point to the MIS project within Philips was a Requirements Analysis (RA) exercise that determined the major requirements of the MIS. The mini-project mentioned previously provided an initial basis for RA. However coupled with this mini-study, RA work had been carried for the first few months of the project using a more structured analysis approach.

Throughout the initial RA, using a waterfall-based SDM, project development was very slow, no sooner had one set of requirements been agreed upon and designed, the need for a major functionality change would arise and the project would be set back. The term 'major functionality' is used in this context to describe a change request that impacts on the overall nature of the designed system. For example within the MIS the main functionality built into the system was based on defined KPIs. Any changes to these KPI definitions would have a dramatic impact on the development of the project and could be termed 'major functionality' changes.

EXAMINATION STATISTICS

The main input to this study was information supplied by the minutes of MIS meetings and documented change requests. Towards the end of the waterfall SDM focus, the issues database was designed and this provided a current log of user requests. Used together with the information stored in existing MIS documentation this provided a source of information for assessing functionality change requests. An investigation was carried out to assess the frequency of these functional changes and determine their effect on the development lifecycle. Figure 5-5 below illustrates changes to requirements specifications throughout the initial RA period whilst using a waterfall SDM such as SSADM.

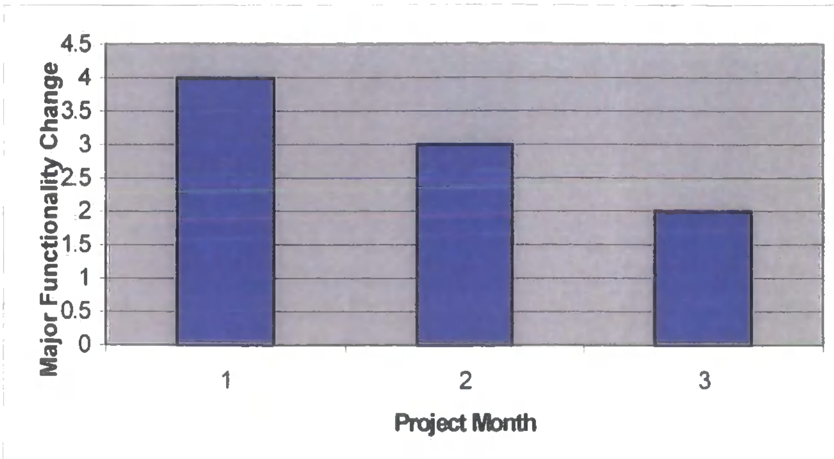


Figure 5-5 - Major Functional Change requests to original MIS specification

The traditional methods chosen tended to be highly restrictive, reasons for which are discussed in Chapter 6 - Analysis of Results. Due to these restrictive characteristics, it was identified that a more flexible method of analysis was needed and hence the Mini Project was initiated, utilising RAD based methods.

5.3.2 The RAD SDM

A second pass at RA was undertaken using the RAD SDM. RAD proved to be useful tool for revisiting original requirements and helped to facilitate several iterations of requirements gathering. RAD was then subsequently applied to the remaining phases within the project.

RAD was used in a number of differing situations to help the MIS team determine project requirements. The two main RA techniques used were management meetings and user sessions. The following sections will demonstrate the results identified from each technique.

5.3.2.1 Management Meetings

The main initiators of the MIS project were the management team with their main requirement being access to long-term strategic decision information. An MIS management team was appointed consisting of 'key' personnel from the Stakeholder diagram illustrated in Chapter 2.

Representatives from the management team were selected based on their involvement with the MIS. Management team members consisted of representatives from several production concerns such as production, finance, quality etc.

TIMING OF MANAGEMENT MEETINGS

Management meetings were used as a technique throughout all stages of the project. As the project progressed, and users became more comfortable with the whole design process, meetings were held on a need by need basis. Figure 5-6 below illustrates the frequency of management meetings throughout the development lifecycle.

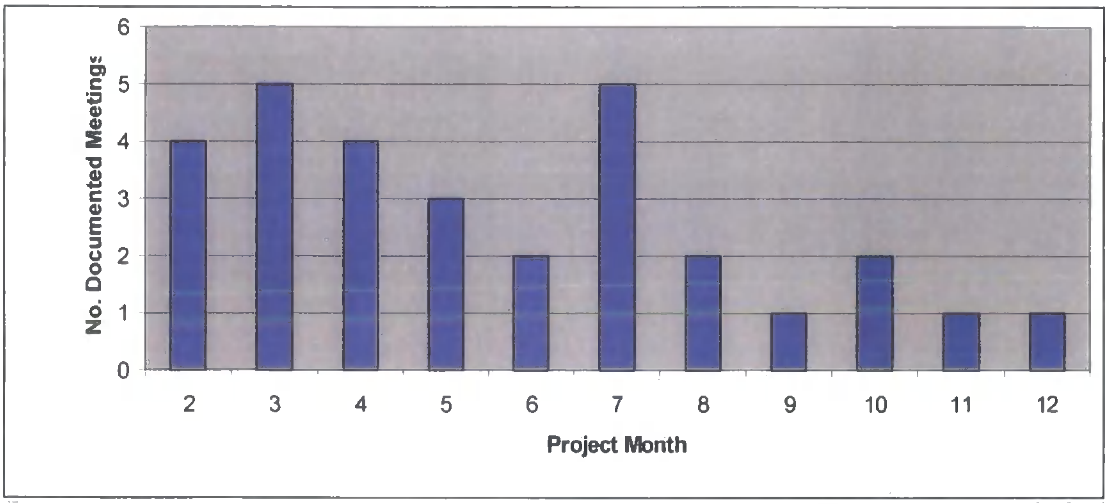


Figure 5-6 - Frequency of Management Meetings throughout project development.

CONTENT OF MANAGEMENT MEETINGS

Within management meetings, before any firm requirements could be discussed, it was important for the MIS development team to be aware of the existing systems in place so that these could be used as a basis for the RA. For this reason the first few management meetings involved discussions around existing manual systems. Initially, the main input to management meetings was a review of current system documentation such as management meeting minutes and early MIS concept documentation. Three members of the MIS development team together with key project stakeholders (approximately eight representatives) then reviewed this documentation to identify areas of weakness.

5.3.2.2 User Sessions

Once the initial requirements for the system had been decided upon, user teams were formed to provide 'end-user' input into the development process.

Members of the user teams varied between lines but typically consisted of a line supervisor and two or three line operators representing each line within the factory. An end user management structure was designed, based upon the overall factory management structure illustrated in Chapter 2. Prior to any MIS design, user sessions usually took place concurrently with management team meetings in order to ensure that everyone had the same understanding of the objectives of the project. As with the management meetings discussed above, user sessions were held throughout the development process and not confined to the early RA stages of the project.

As the project evolved, user sessions were held on a need by need basis and ranged from training sessions to one to one 'interviews'. Figure 5-7 below illustrates the frequency of user sessions throughout the project.

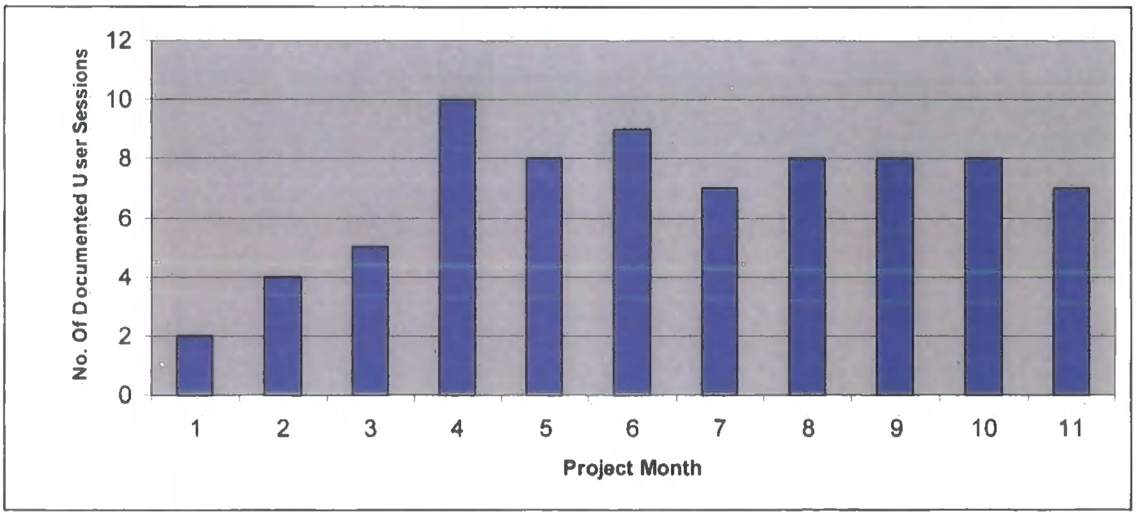


Figure 5-7 – Frequency of User sessions throughout the MIS project

CONTENT OF USER SESSIONS

Due to the highly interactive nature of user sessions the content and objectives of these sessions ranged widely. Sessions could incorporate training, discussions around PC availability on the lines or design of screens and data collection sheets. The user sessions however tended to be more reactive rather than proactive and were usually held when an issue occurred with the system. User sessions tended to be slightly less structured than management meetings due to their content and interactive nature.

Once all team members, users, management and the development team, were agreed on the requirements of the system, a set of 7 KPI definitions were devised to encapsulate the requirements. Chapter 2 discusses the identified KPIs.

Throughout the development process, management meetings and user sessions were held not only to assess project progress but also assess the suitability of the system being designed. Throughout the development of the MIS the main outcomes of management meetings and user sessions tended to be change requests.

In order to assess how RAD coped with these change requests, analysis was carried out to examine the nature of these changes and their frequency. Not only was analysis of the frequency of change requests carried out but also an examination of the nature of the requested changes. This analysis provided information about the system being developed i.e. the progress of development could be running well ahead of schedule but does the system meet the users' needs? Using CPFs and the MS Visual Source Safe log together with the issues database, analysis was carried out into these changes. Figure 5-8 below illustrates the comparison between 'functional' and 'cosmetic' change requests received throughout the project. The term 'cosmetic' in this context refers to a change in the appearance of the system, a change that typically has no major impact on the system.

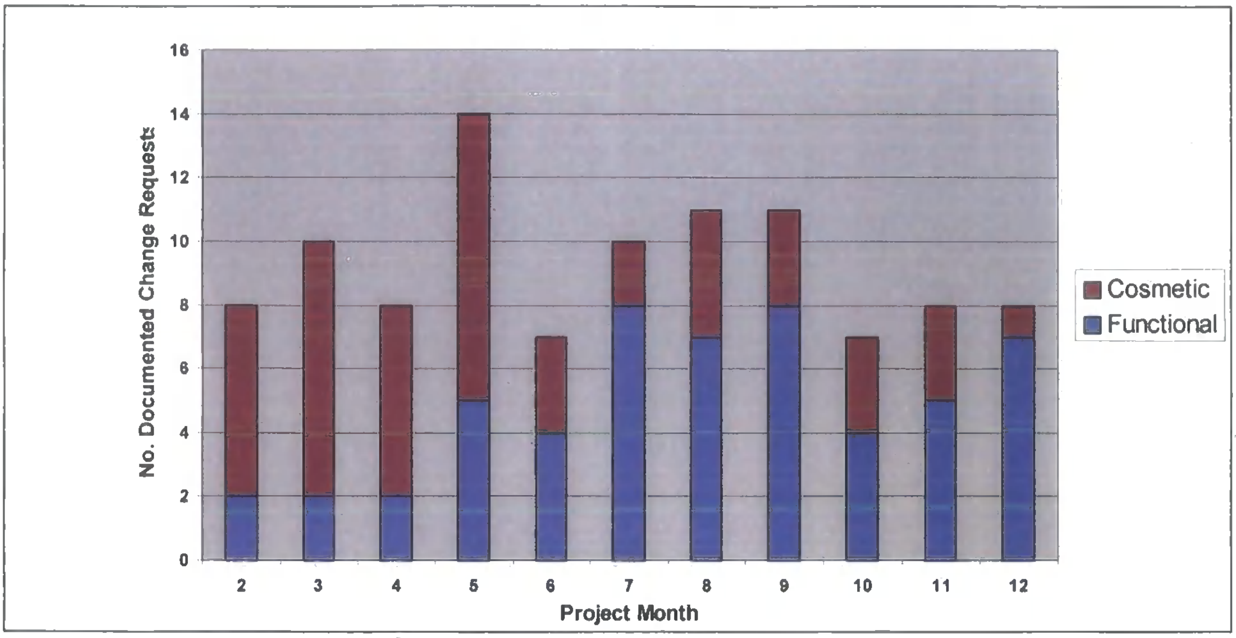


Figure 5-8 – Cosmetic versus Functional Change Requests

The concurrent MS Project Plan was then used to analyse the overall time impact that the change requests had on the development process.

5.3.3 SUMMARY

This section has predominantly examined the results provided by the application of both RAD and a waterfall-based approach with an emphasis on requirements gathering. With reference to Figure 5-1 it can be seen that for the first month of the project a traditional approach was applied. Due to the short timescale of the project, the waterfall SDM was not deemed satisfactory and hence from month two a RAD SDM was adopted.

5.4 CHANGE MANAGEMENT

This section will document how RAD was used to manage the organisational changes imposed by the MIS.

As part of the 'change management' consideration the two main techniques applied were training sessions and user guide documentation. However, in order to assess what effect other 'change management' techniques would have had on the overall result additional research was carried out using these techniques once the project was complete. Examination of these 'human' aspects allowed the development team to become more aware of their customers and hence target development to meet user needs.

5.4.1 TRAINING

Using the methods described in Chapter 4 the following results were provided by the MIS Training Log.

Figure 5-9 illustrates the number of employees that received MIS training on each line. Figure 5-10 illustrates amount of time accrued in training across each line. The graph provided in Figure 5-11 illustrates the order of MIS implementation and the project month in which MIS was implemented on each line. Figure 5-12 illustrates the timing of training in comparison to MIS implementation.

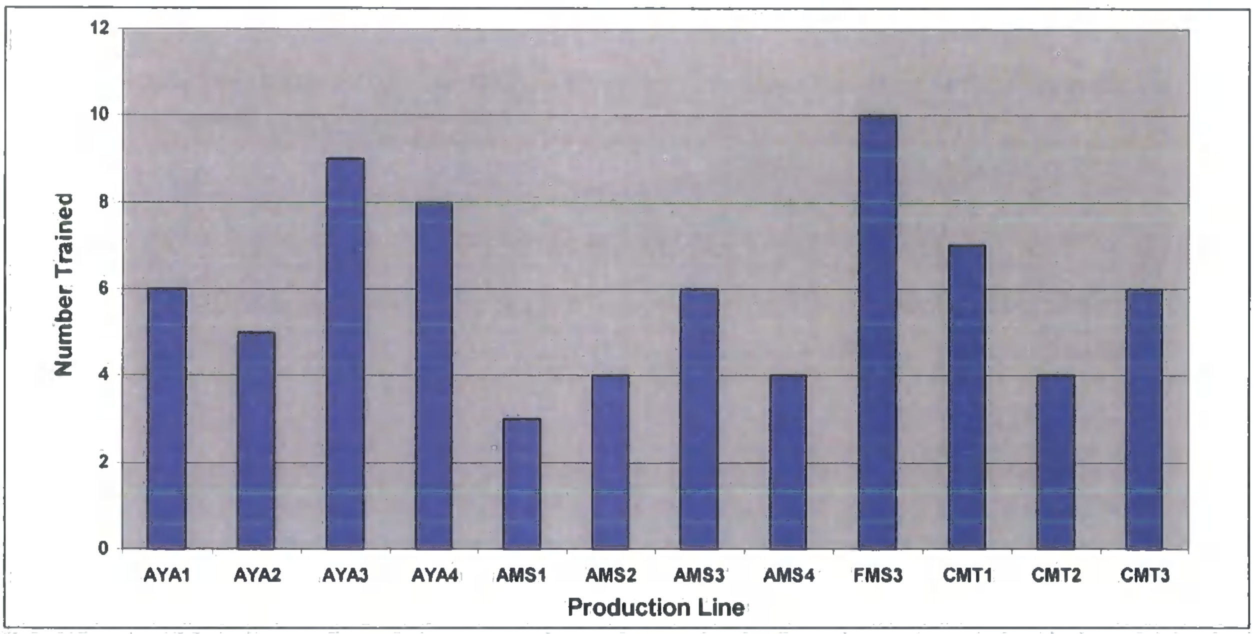


Figure 5-9 - Number of employees trained in MIS across each line

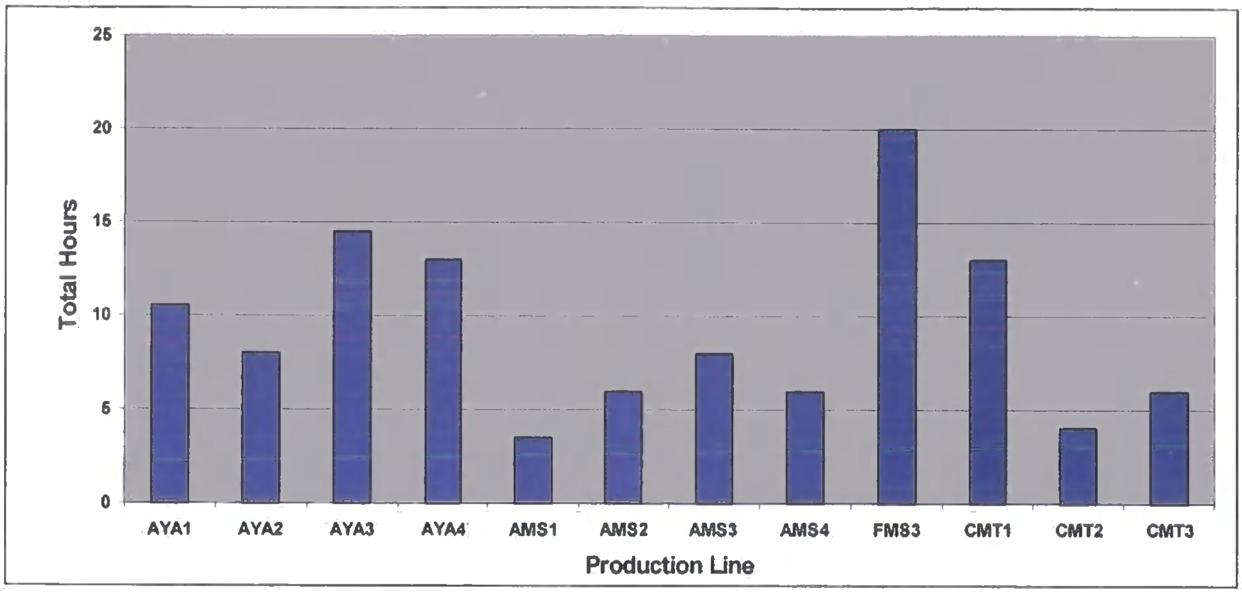


Figure 5-10 - Total Training Time Across each Line

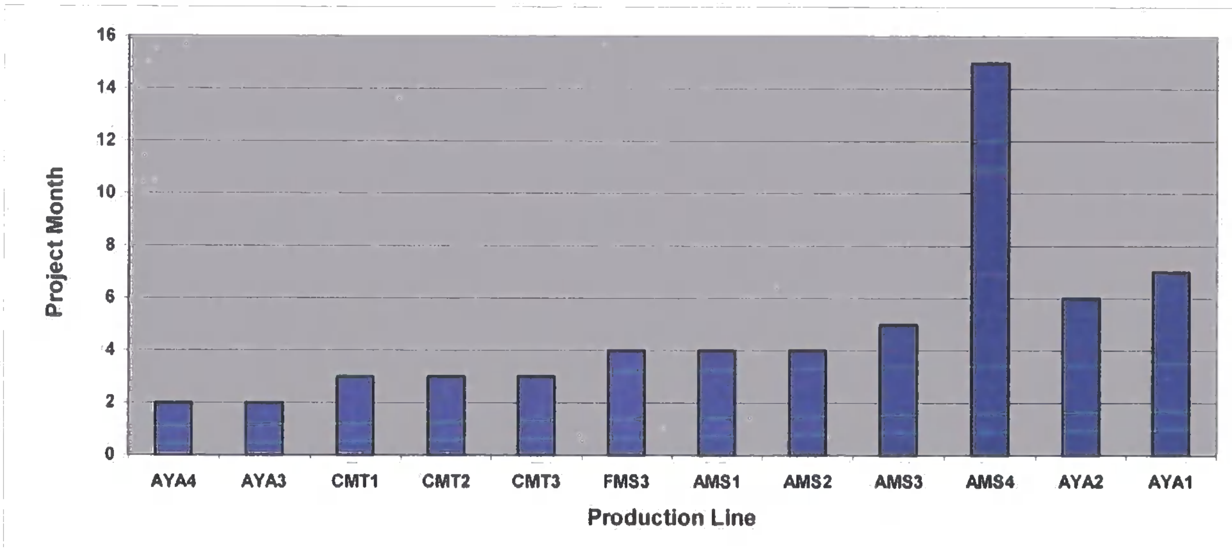


Figure 5-11 – Order of MIS Implementation

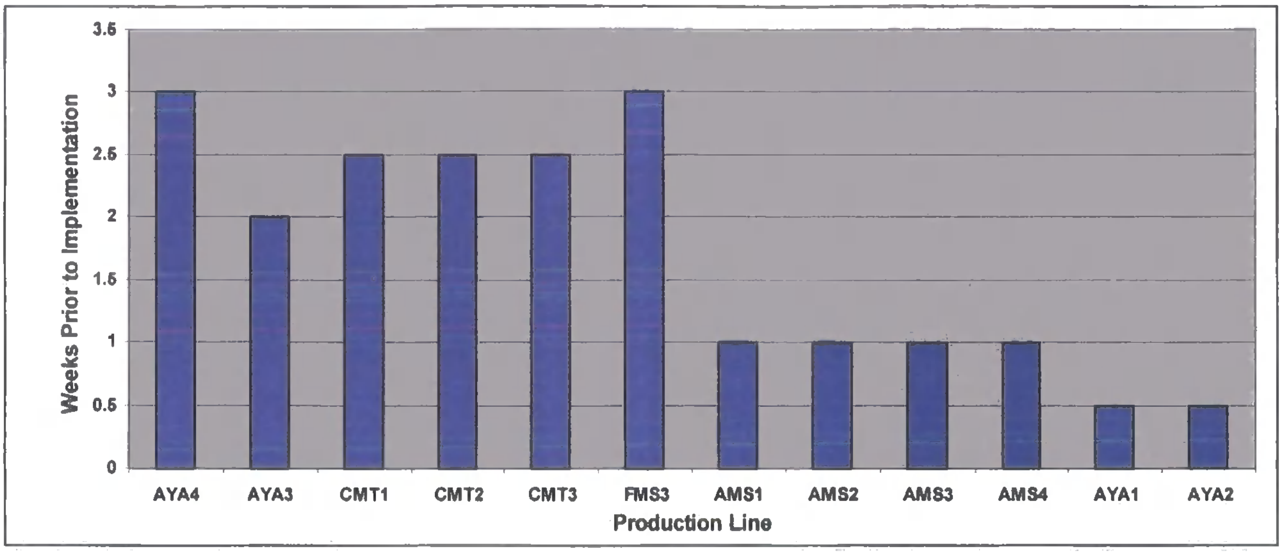


Figure 5-12 - Delivery of MIS Training prior to implementation

5.4.2 MIS USER GUIDE

Using the methods described in Chapter 4, the following results were obtained from the issue of the MIS User Guide. Due to the frequent changing of user requirements, the MIS system evolved through several versions. Each time the MIS changed, a new version of the MIS user guide was required. Figure 5-13 below illustrates the issue of user guides over the lifecycle of the project. With reference to Figure 5-13, in month 1 the first eleven user guides were issued. Any issue of user guides after month 1 represents re-issued copies of the original user guide.

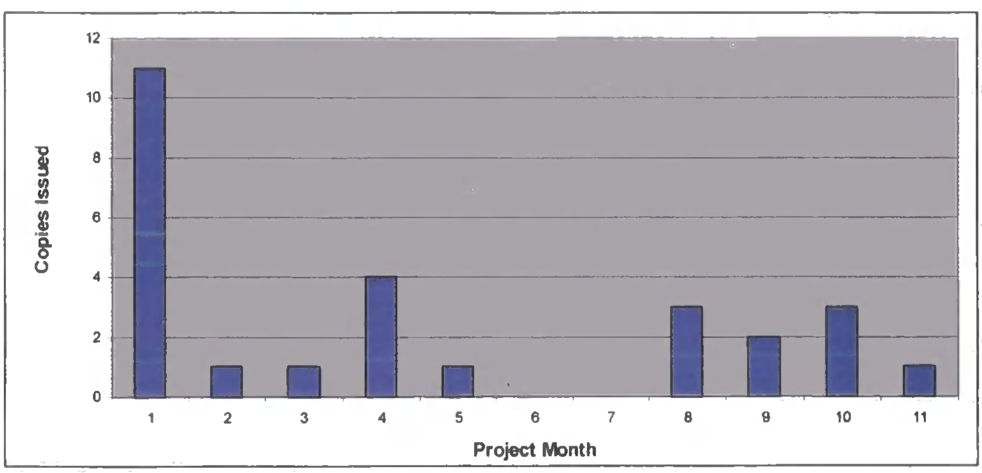


Figure 5-13 – The issue of the MIS User Guide

5.4.3 Personality Mix Study

As described in Chapter 4 the personality mix study was applied once the project was complete. This technique was applied to assess the characteristics of different people in the group and was used in future stages of the project to assist the development team in managing the end users appropriately.

Examination Statistics

The main input to this study was the data entry investigation documented in section 5.5 below. The data entry investigation examined the speed and accuracy of data being entered into the MIS system two months after its initial implementation. This investigation helped the development team assess issues such as training problems and general issues associated with the MIS process. A more detailed explanation of the data entry investigation can be found in section 5.5. Further analysis was also carried out using interviews. Approximately twenty interviews were conducted; interviewees were selected based on their involvement with the MIS project. Participating interviewees were taken from several production concerns including engineering (20%), line operators (50%) and line managers (30%). Out of the employees interviewed 80% had direct involvement with the MIS project.

Figure 5-14 illustrates the mix of end user personalities within the Philips organisation taking into consideration both management and operational end users.

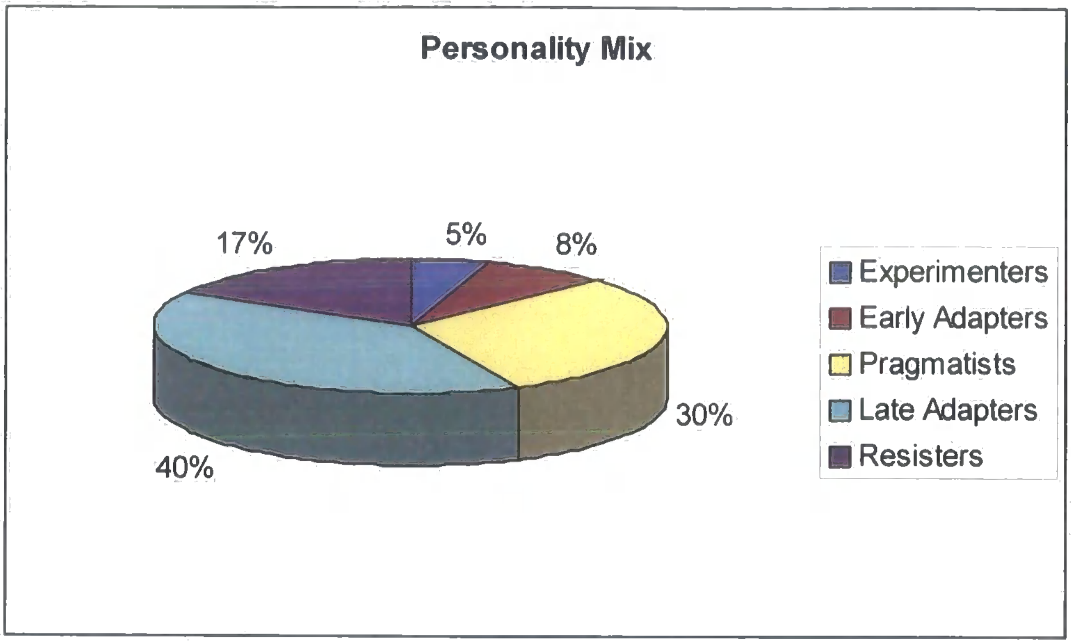


Figure 5-14 - Mix of end user personalities within Philips

5.4.4 Motivation of Users Study

The second 'additional' user analysis technique applied once the project was complete was an assessment of the end users motivation. Using the methods described in Chapter 4 a study was carried out examining the main motivational factors that would 'drive' the end users of the MIS. In order to concentrate this study purely on MIS issues, users were interviewed in relation to their wishes of the MIS not their overall motivation factors within the organisation.

Examination Statistics

The main source of information was achieved through interviewing both operational and managerial end user groups. Five representatives were selected from each user community and interviewed with respect to their motivations for using the MIS system. Approximately 12 interviews were conducted over a two-week analysis period. Additional information was also taken from the results of the TCS mini-project. Figure 5-15 illustrates the main motivations for the use of MIS within the operational end user community. Figure 5-16 illustrates main motivators of the managerial end user community.

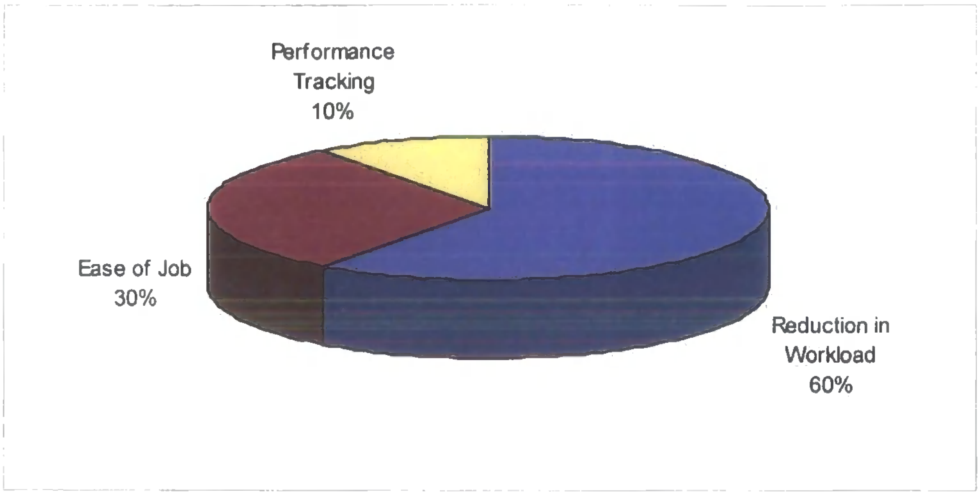


Figure 5-15 - Motivational factors provided by the MIS for the operational end user community.

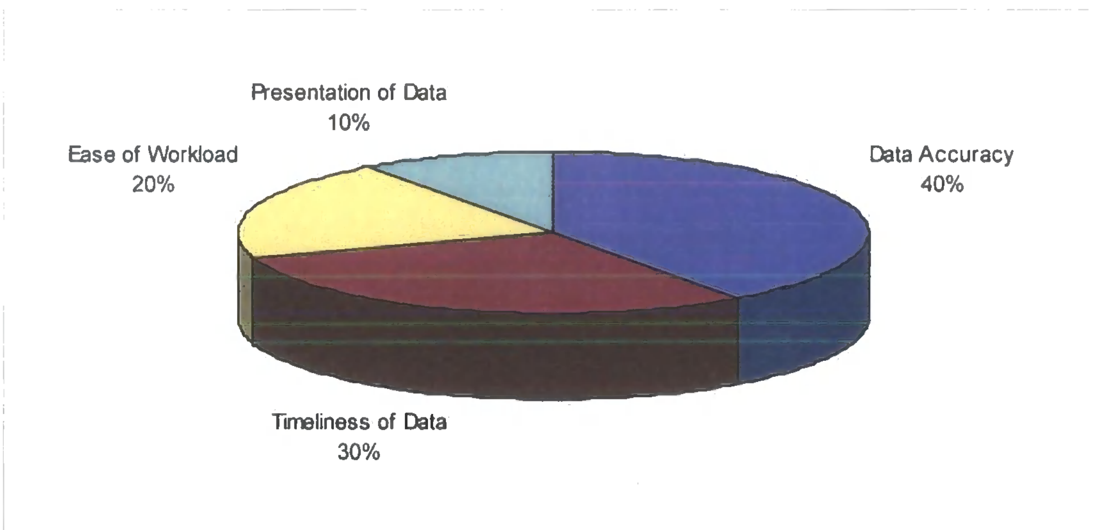


Figure 5-16 - Motivational Factors provided by the MIS for the Managerial end user community.

5.4.5 SUMMARY

The above section has helped to identify that whenever changes to working practices are introduced, an in-depth study of 'human factors' should be carried out prior to project initiation. This is true of all projects regardless of objectives. The results provided have helped to illustrate the various 'change management' techniques that were applied throughout the project and have also helped to identify techniques that could be applied in the future.

5.5 Data Entry Investigation

The study was split into three main areas. Firstly an examination of the percentage of data entry across each line was carried out. This investigation allowed the development team to assess exactly what percentage of the required data was being entered across each line. This then highlighted any problems on particular lines such as the lack of available data, lack of required personnel etc., which could be addressed accordingly. These tests were carried out over a number of weeks, two separate data entry investigations were carried out; the first in project month 3 and the second in project month 8.

Figure 5-17 below illustrates the total number of records being entered across each production line. This provides the reader with an indication of the varying information requirements across each line.

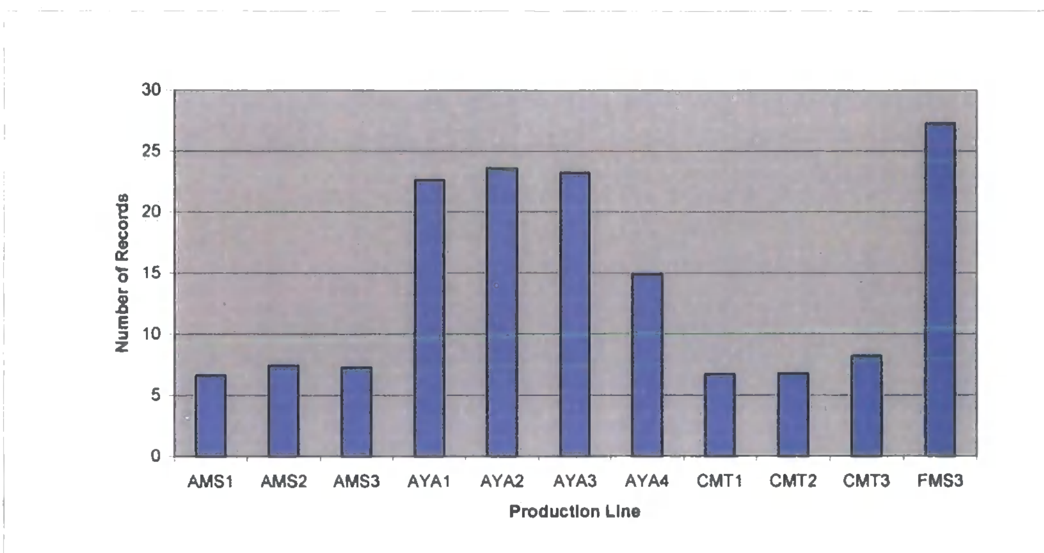


Figure 5-17 – Total Number of MIS records entered by line.

Figure 5-18 below illustrates the percentage of the required data being entered into the MIS system over several months. These graphs illustrate how, over time the users' management of the MIS varied.

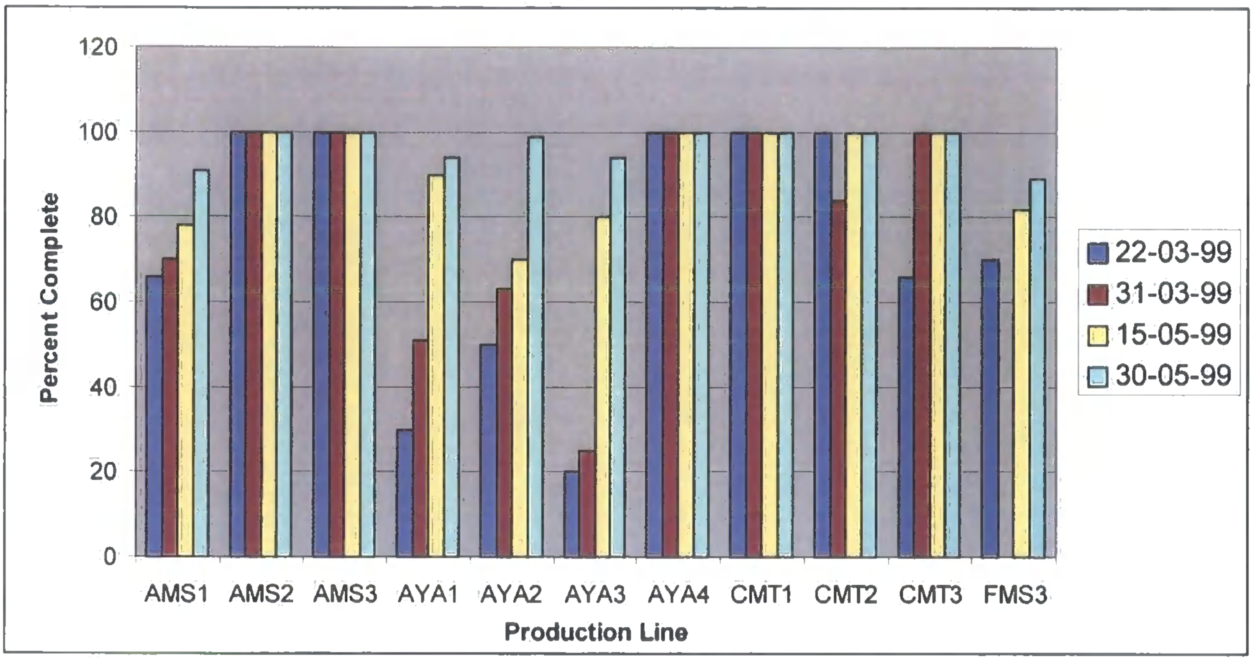


Figure 5-18 – Data entry completion throughout project month 6 and 8

Investigations were also carried out assessing how long it was taking each shift on each line to enter the required data. A comparison of the total time taken to enter all line data and the total time taken to enter one record was carried out. This helped the development team assess if a particular area of the MIS was causing problems. Figures 5-19 below illustrates one example of the data entry times per record and total shift time on a particular production line. This investigation was carried out in project month 8.

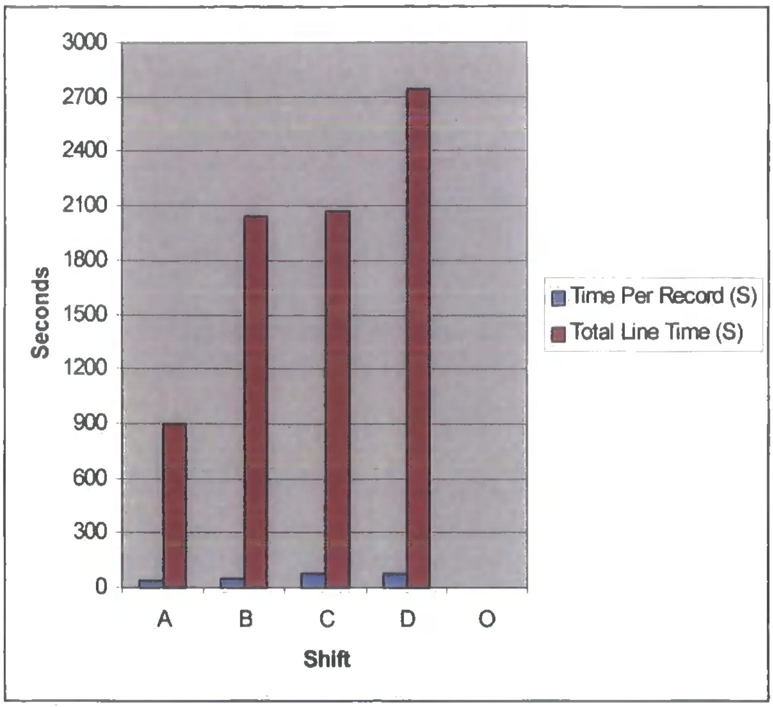


Figure 5-19 – Data Entry Times AMS1 project month 8

A further set of tests involved an analysis of the accuracy of the MIS data being entered. Various data recording methods existed around the factory for the measurement of performance; some of these methods were used in parallel with the MIS in order to compare the accuracy of data. These tests were carried out three months after the initial MIS implementation across the factory (project month 10). Figure 5-20 illustrates the comparison between MFGPRO, the Scrap database and the MIS with reference to the figures recorded for packed volume.

Results from existing systems were compared against the data output from the MIS to determine the accuracy of the MIS system.

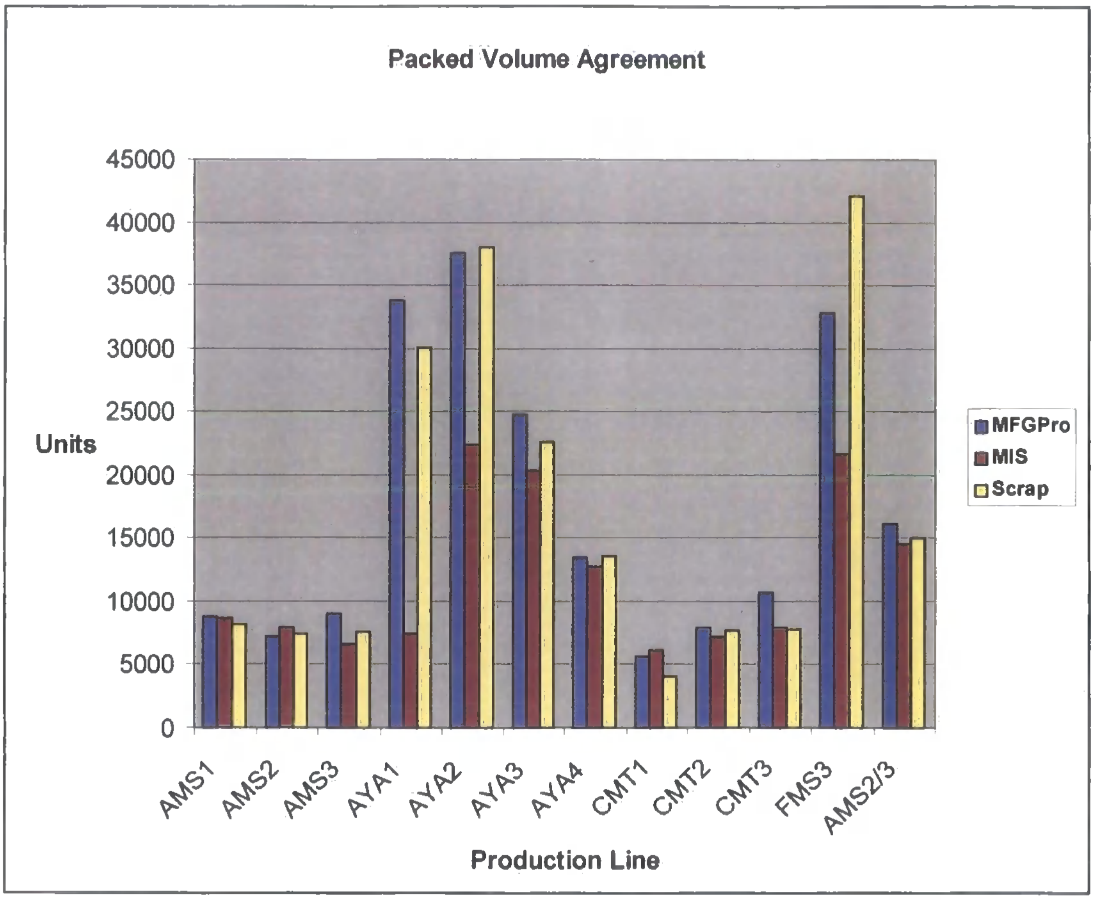


Figure 5-20 – System comparisons for Packed Volume project month 10

5.5.1 Data Entry Summary

As described in Chapter 4, the main objective of the data entry investigation was to assess the effectiveness of end user management, indicating to the development team the users attitude towards the implemented system. This study helped to graphically highlight any problems that users were encountering with the MIS and provided the development team with a useful end user management tool.

5.6 Project Management

This section will document how the RAD approach coped with numerous changes to initial requirements. Three main methods were used to measure this; the issues database, change proposal forms (CPF) and the visual source safe log. The information that follows will provide an illustration of the results obtained from each method, additional examples of which can be found in Appendix D.

5.6.1 MS Visual Source Safe

Due to lack of technical support and understanding of Visual SourceSafe this tool was only used for the first three months of the MIS development. Unfortunately some configuration problems were encountered and due to lack of expertise within this area, and time restrictions, SourceSafe became unusable within the MIS project

Figure 5-21 below illustrates the frequency of implemented changes to the MIS code within the first three months of the project initiation. The SourceSafe tool was also used as part of the analysis illustrated by Figure 5-5.

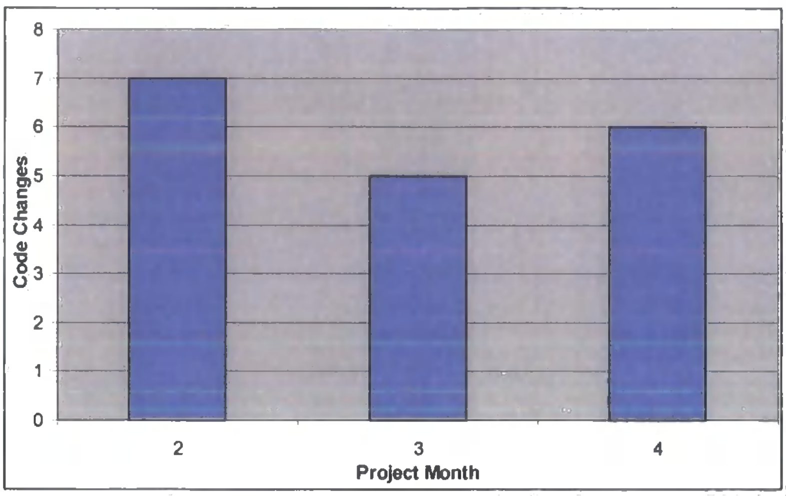


Figure 5-21 - Code changes within three months of project initiation

Analysis of the SourceSafe tool also provided results on the reuse of code modules within the MIS project. Within the first three months of development, Figure 5-22 below illustrates the percentage of code modules that were re-used either as part of further modules or separate code modules within other applications.

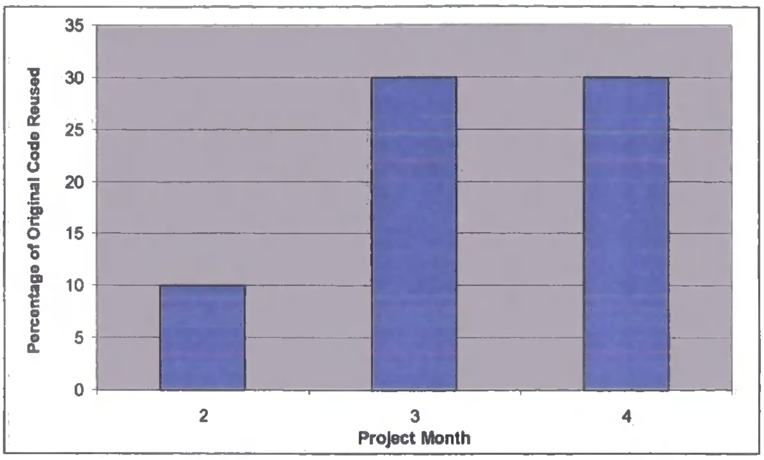


Figure 5-22 – Percentage of re-use within MIS project.

5.6.2 THE ISSUES DATABASE

This method of project management was used from project month 3 and for the remaining duration of the project. As the project progressed it can be seen with reference to Figure 5-23 below that the recording of issues seemed to reduce.

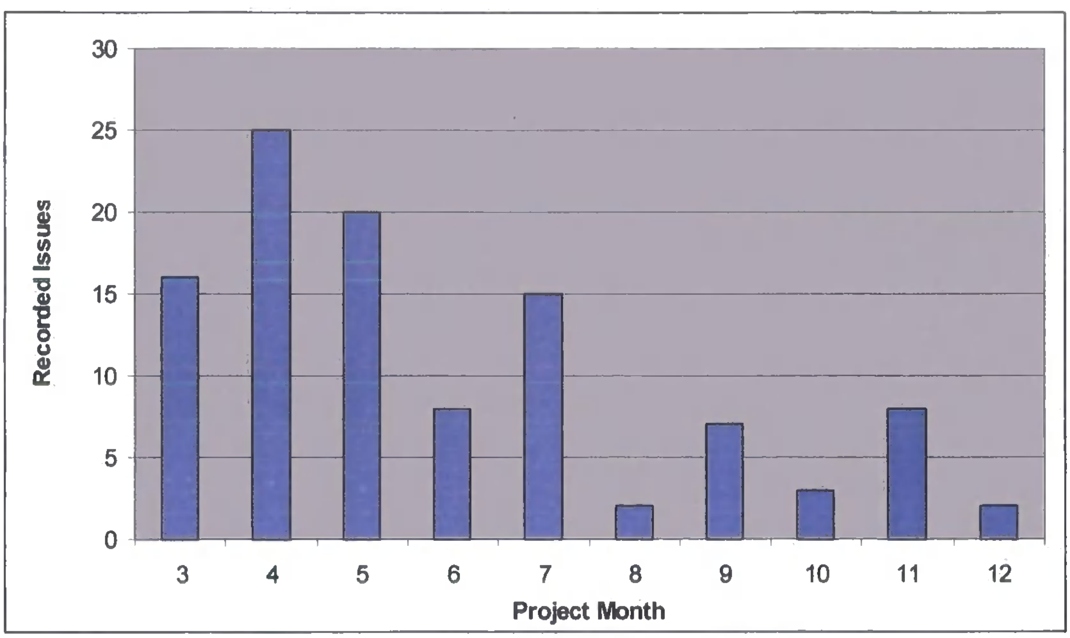


Figure 5-23 – Recorded Issues throughout project lifecycle

The issues database consisted of a backend database created in Microsoft Access and a front-end interface also created in Access, an illustration of the interface to the issues database is provided in Chapter 4. Figure 5-24 below illustrates an example of some of the issues raised.

ID	Issue	DateRaise	Raised By	Action By	Resolution
1	On change overs when operators have other jobs to do, where does this down time get logged?	26-Nov-98	J.D.	S.D.	Possibly add a comments field in the operator hours pop up window.
2	Reasons for scrap for standard wire are not always known exactly.	26-Nov-98	J.D.	A.W.	Only the primary reason should be entered into the scrap field. Unless an improvement project requires more detail
3	On frame and Line winding machines they are rarely zeroed between shift changes. This means that P1 faults can often be carried across shifts.	26-Nov-98	J.D.	S.D.	Talk to three supervisors to ensure that winding operators are trained and that skill sheets are updated.
4	When measuring scrap for frame and line winding machines do we measure a total scrap for both machines added together?	26-Nov-98	J.D.	L.M.	Redesign the scrap entry form to include a cell name and a note 'scrap excluding scrap from winding machine'.
5	Line Engineers are already collecting some of the data that is to be entered into the MIS, where does this information go?	26-Nov-98	J.D.	S.D.	Talk to Line engineers to find out what is being collected twice. Then decide which information they still wish to collect.

Figure 5-24 – Typical Issues raised throughout the project lifecycle

5.6.3 Project Management Summary

This section has helped to illustrate the differing methods of project management that were applied throughout the MIS project. Project management within the project was assessed to determine whether the RAD SDM, being highly re-iterative still had the capability to provide a controlled development environment. The Visual Source Safe tool, although only used for a short period of time, provided version control and a method of reusing code modules within MIS. The main methods of project management were the CPF and the issues database, both providing a flexible method of recording and monitoring various change requests throughout the lifecycle of the project.

5.7 PROTOTYPING

One of the main tools used throughout the development process was the prototype. This tool was chosen for many reasons, an explanation of which can be found in Chapter 4 – The Development Process.

As described in Chapter 4, all MIS prototypes were developed as fully coded Access applications. After a release, if the prototype needed amendment changes could be made to the original prototype to enhance its capabilities. Since each prototype contained executable code, once a change had been made, the code behind the prototype could be compiled and the prototype re-released for testing.

Using the RAD SDM, due to the iterative nature of this methodology, several prototype releases were made. Figure 5-26 illustrates the final MIS product. The graph provided in Figure 5-25 illustrates the number of prototype releases made throughout the project lifecycle and the stage at which they occurred within the project lifecycle. This graph is intended to illustrate that prototyping was not restricted to the early stages of the project lifecycle. Prototyping was used as a tool throughout the entire development lifecycle, it was not restricted to requirements analysis. Once the prototype had been released it was then built upon and re-released until all requirements had been satisfied. The final fully working system was delivered and in use by project month 10.

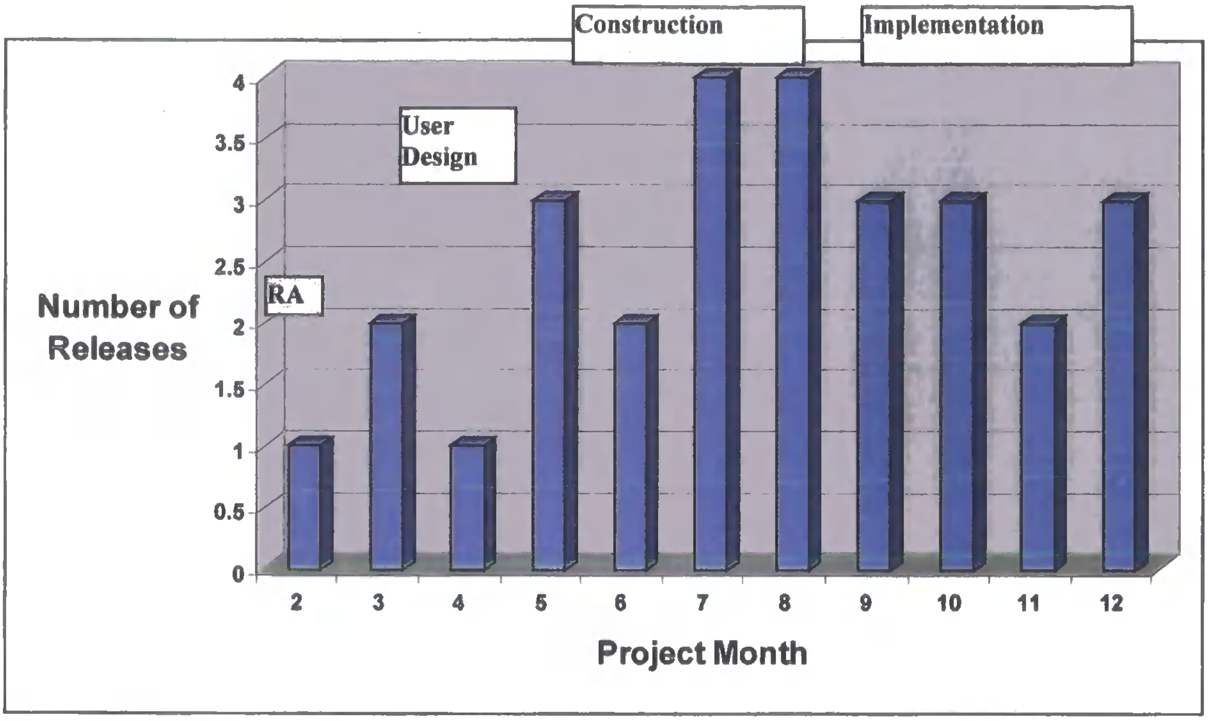


Figure 5-25 – Number of prototype releases

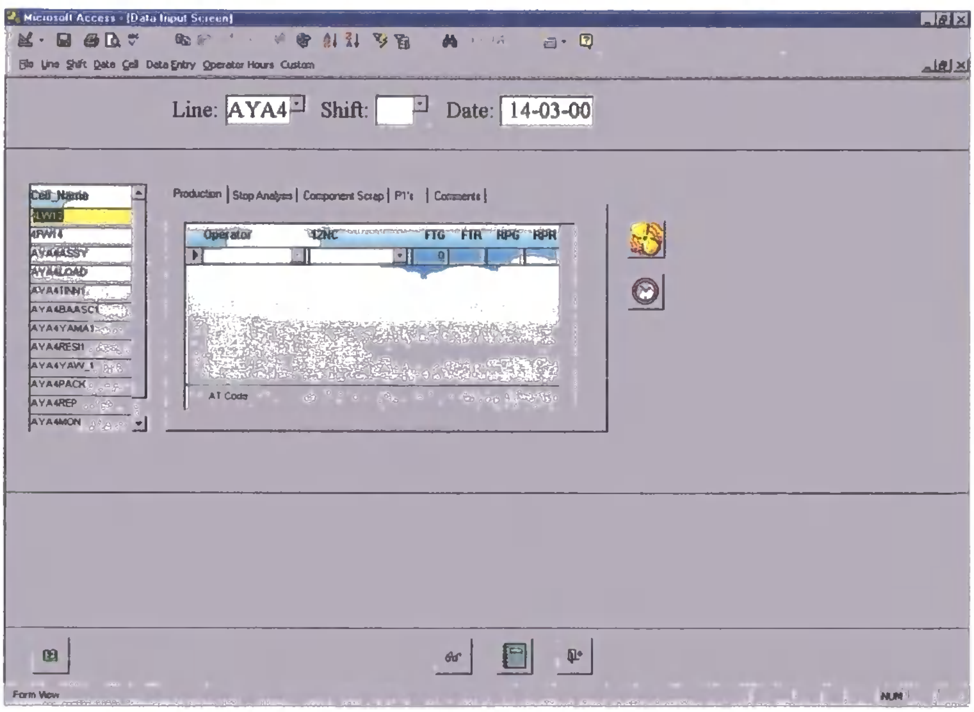


Figure 5-26 – The Final MIS system

5.7.1 Prototyping Summary

In summary, prototyping through RAD provided the development team with its main RA tool. Prototyping was used within all stages of the RAD SDM and hence requirements were continuously revisited helping to ensure that only current functionality was built into the system.

5.8 Summary

This chapter has helped to demonstrate the various results obtained through the application of the RAD SDM. It can be seen that many of these results do focus on the human aspect of the RAD methodology and help to demonstrate the heavy 'people' focus that this approach has. Through this 'user' focus, these results also help to highlight the importance of understanding the final customer. The results provided under the prototyping section have demonstrated that RAD is very iterative and not confined purely to requirements gathering.

CHAPTER 6

ANALYSIS OF RESULTS

6.1 INTRODUCTION

This chapter aims to provide the reader with an analysis of the results identified in Chapter 5. The results identified will be discussed and evaluated with respect to the documented characteristics of RAD listed below:

- **High Quality Solutions**
- **Increased Development Speed**
- **Increased User Support**
- **Frequent Delivery of End Products through Prototypes**
- **Reuse**
- **Change in Culture**
- **Heavy People Focus**
- **Integration**
- **Choice of Tool**
- **Creeping Functionality**

Finally an assessment will be provided on the choice of SDM used. This will examine if the RAD SDM was the most appropriate in the development of the Philips MIS.

6.2 BENEFITS OF THE RAD APPROACH

6.2.1 High Quality Solutions

In Chapter 3 one of the quoted benefits of RAD was its ability to produce higher quality solutions through meeting the users' needs more effectively. This section will analyse the results of how RAD was applied throughout the RA phase examining if the application of RAD lead to higher quality.

One contribution to quality is the fact that RAD allows requirements to be revisited. Figure 5-9 illustrates that six months into the project, requirements were still being revised, RAD allowed the development team to ensure that only the most current requirements were being developed without too much impact to the timescale of the project. Stapletons paper quotes that research has shown that during a one-year project, 10% of requirements are likely to change **[STAPLETON, 1995]**. If using standard traditional analysis methods, this usually means that a lot of effort in quality control is actually targeted on the wrong systems in terms of what the business requires. In comparison to this RAD provided a higher quality deliverable by concentrating on key requirements before proceeding with lengthy development.

As illustrated, a key contribution to building a product of high quality is the ability to accurately satisfy user needs. This requires close co-operation with the end user group throughout the project lifecycle. Through RAD, the ability to review how the users were using the system helped to ensure that user needs were being met even after implementation of the final product. The information provided by the various data entry investigations in section 5.5 helped to identify the following issues:

- Firstly the levels of training requirements on each line were addressed. With reference to Figure 5-18 it can be seen that only four out of the eleven lines examined were entering one hundred percent of the MIS data required. Examination of line AYA3 suggests that this particular line at the time of analysis were experiencing problems with the MIS system. Two months later the completeness of data being entered into the MIS had increased. Data entry investigation helped to determine if RAD had taken these issues into consideration. In the case of the MIS study, these human issues had been addressed within RAD as illustrated by Figures 5-10 and 5-11 under the User Training section.

- Date entry investigations allowed the development team to assess the appropriateness of the delivered system. One of the main acceptance criteria for the MIS system was that MIS data must be collected and entered within one hour of the shift end. Figure 5-18 helped to highlight that in month 8 four, possibly five of the shifts on certain lines were not able to meet the set criteria with the AYA3 line causing the most concern. In this case system requirements were re-examined in order to address the problem.

- With reference to Figure 5-20 comparisons can be made between MIS and existing systems. The first apparent result is that when assessing the figures reported for the packed volume, the existing MFGPRO figure differed in most cases to the reported Scrap database figure. After further examination it was identified that there were discrepancies within the Scrap database. This result helped to illustrate that a more standard method of data recording was required in order to standardise the performance figures that had been identified through RA across the factory.

In summary, the assessment of quality provided by RAD has demonstrated that in order for a higher quality deliverable to be produced, RAD must be applied in a flexible manner. With reference to the figures in Chapter 5, as the project progressed RAD had to be flexible enough to accommodate these findings and address any problems identified. The results helped to illustrate that due to the increases in data accuracy, reduction in time to enter the data and increases in the completeness of data, that RAD did allow for improvements to be made. RAD, through its iterative nature and flexible framework supported the redefinition of requirements and re-work of solutions, thus helping to provide the customer with a system that met their needs.

6.2.2 Increased Development Speed

A further quoted benefit of RAD is a reduced development lifecycle. Once analysis had been carried out on the number of change requests received examination was then carried out into RAD's effect on the overall development lifecycle.

Unfortunately the only comparison that could be made between RAD and a traditional approach had to be done within the RA stage. The results provided throughout the RA stage can be misleading since in any project, requirement requests and changes to requirements are usually higher in the RA stage than in any other project stage. Hence a true comparison of the effects of RAD could only be taken when examining progress made against documented project plans. For this reason analysis of the comparison between the traditional approach and RAD has been omitted.

Using the RAD SDM for continuous RA helped developers to gain a list of requirements in a short timescale, ensuring that the most current functionality was built into the system. The development team initially focused on high-level requirements and once these were agreed, a more detailed approach could be taken to improve the system. Traditional approaches tend to place more focus on getting a system to satisfy all requirements before development begins **[STAPLETON, 1995]**. This tends to lead to long time scales and requirements that do not match business needs. RAD facilitated quick development based on high-level requirements, an initial MIS system was built early on in the project lifecycle and then refined as further requirements were identified.

Prior to project initiation a set of high-level MS project plans were produced detailing the various project activities that were required and the proposed timescales. The original timescales proposed were based on past projects using more traditional SDM techniques. With reference to the MS project plan shown in Figure 6-1 it can be seen that when using RAD the planned timescales for development were reduced. The graph provided illustrates the proposed plan for the design stage of the project. An illustration of the overall project plan can be found in Appendix E.

ID		Task Name	Duration
1		**1 MIS Phase 1 Design (planned)**	**108 days**
2		**2 MIS Phase 1 Design (actual)**	**76 days**
3		**2.1 Define Requirements**	**3 days**
4		2.1.1 Define Outputs (planned)	3 days
5		2.1.2 Define Outputs (actual)	2 days
6		**2.2 Define Inputs**	**13 days**
7		2.2.1 Define Equations (planned)	8 days
8		2.2.2 Define Equations (actual)	5 days
9		2.2.3 Define Base Data (planned)	7 days
10		2.2.4 Define Base Data (actual)	5 days
11		2.2.5 Check Base Data Availability (planned)	3 days
12		2.2.6 Check Base Data Availability (actual)	2 days
13		**2.3 Simulate System**	**11 days**
14		2.3.1 Define Test Line (planned)	2 days
15		2.3.2 Define Test Line (actual)	3 days
16		2.3.3 Design Test Bed (planned)	5 days
17		2.3.4 Design Test Bed (actual)	3 days
18		2.3.5 Design Scenarios (planned)	3 days
19		2.3.6 Design Scenarios (actual)	3 days
20		2.3.7 Test Senario Outputs (planned)	3 days
21		2.3.8 Test Senario Outputs (actual)	2 days
22		**2.4 Document Procedures**	**27 days**

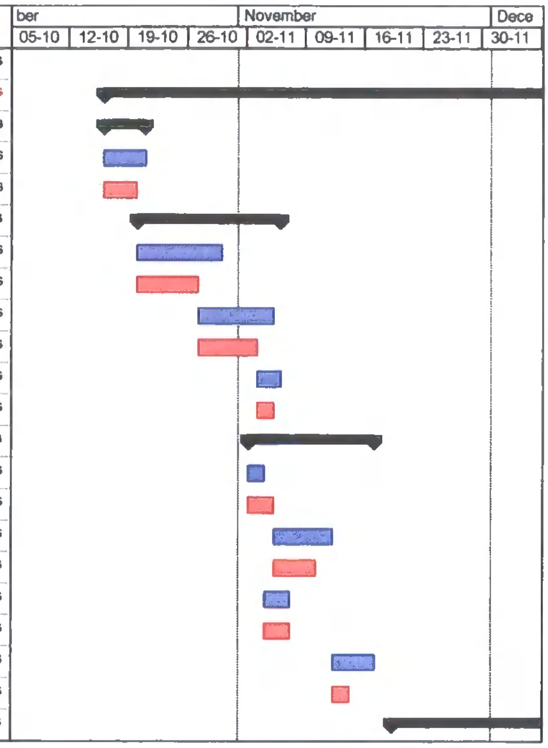


Figure 6-1 – MS Project Plan for design phase of MIS

6.2.3 Increased User Support/Change Management Control

The following analysis can be made:

- From a management end user perspective Figure 5-6 illustrates that initially the users participation was very good. For the first four months of the project a management user meeting was held on average at least once a week. However as the project progressed these management meetings tended to be more reactive than proactive and were only really held when a problem arose with the MIS.

- Once the initial requirements had been agreed upon user involvement did tend to be overlooked and empowerment only really happened on a need by need basis. It seemed that after the initial four months, management users only really became closely integrated with the project whenever a major decision was required.

- Figure 5-7 illustrates that within the first four months of project initiation there was relatively little operational end user involvement. The main reasoning behind this was a large proportion of development time was initially taken up on deciding the exact management level user requirements. Operational involvement during this period remained on reactive level. However once the system became a factory wide concern, more and more operational end users became involved in the project and the level of operational user involvement started to increase. With respect to operational end users it can be seen from Figure 5-7 that this user involvement remained high for the remainder of the project lifecycle.

Analysis was also carried out to examine how RAD helped to manage the impact of change. The information provided below is based on the various change management techniques that were applied either as part of the main project or once the project was complete.

One of the first 'User' considerations made was how the development team were going to manage the end user community. Figure 2-4 illustrates the management structure that the end user structure was based upon within the MIS project. The stakeholder diagram provided in Chapter 2 illustrates that a wide variety of personnel were involved in the MIS. For this reason it was decided to split the end users of the system into two groups, operational end users and managerial end users. As analysis began it became apparent that within these two user groups several classes of user were also present. Assessment of end user management concluded that in order for user support to be successful overall management should be divided between the key personnel involved.

Utilisation of this end user management structure provided the development team with the following benefits:

- Allowed developers to understand who the end users were and how best to manage their expectations.

- Allowed the project team to sub-divide responsibilities, which helped to promote the empowerment of users throughout the project.

- Users were involved from an early stage within the project. By giving end users responsibilities they no longer felt that the MIS was purely an IT project, designed by IT and then forced upon them. Since users were part of the decisions made they felt part of the project from the start.

Within a project such as the MIS it is clear that a defined method of end user management is crucial to the success of the project. This study enabled such a method to be agreed upon and in turn helped facilitate the management of both managerial and operational end users. It could have been decided that the development team take on all management responsibilities for the end user however within the time constraints and taking into consideration the resources available, this was discarded as a viable option. This study helped to highlight the importance of establishing exactly whom the customers of the system are and how to get the best from them.

Another important aspect of RADs 'User Focus' is that of user training. With reference to Figure 5-9 it can be seen that over the project lifecycle approximately 72 people were trained in the use of the new system. This training took approximately 113 hours or the equivalent of 14 days worth of training based on an 8-hour day. Figure 6-2 below illustrates the total hours of training that each employee received compared to the length of time that training was delivered prior to the MIS being implemented. Using the results provided in Chapter 5 the following analysis can be made with reference to user training:

- As can be seen from Figure 5-10, training needs across lines did vary, it was important to identify that training needed to be flexible across each of the production lines. It became obvious that training needed to be tailored to each specific line in order for everyone's needs to be addressed. RAD did allow the development team to identify this need for flexible training.

- Through the use of a prototype and early involvement of the end users, RAD facilitated the use of continuous training as the project progressed. Also due to the flexible nature of RAD training needs could be continuously tailored to meet the needs of the users.

- Figure 5-12 illustrates that some training was delivered a long time before the MIS system was implemented. Early delivery of training did cause some problems such as users forgetting the training and not fully understanding the system due to the long delay between training and implementation of the final system. In these cases further training sessions had to be delivered in order to keep users on these lines fully up to date with the system.

- Figure 6-2 illustrates that generally in cases where training was delivered several weeks before MIS implementation this resulted in an increase to the amount of training required, FMS3 and CMT1 are good examples of this. There are some exceptions to this however such as CMT2, even though training was delivered 2.5 weeks before implementation only 1 hours worth of training was delivered to each employee on this line. This graph helps to highlight the need to examine the timing of training sessions as well as the content. One method of overcoming this problem would have been to 'timebox' the training sessions on each line. Unfortunately since research into RAD took place in parallel with MIS development this method, although well documented as part of the RAD SDM was seldom used. Timeboxing is explained in more detail in Chapter 3.

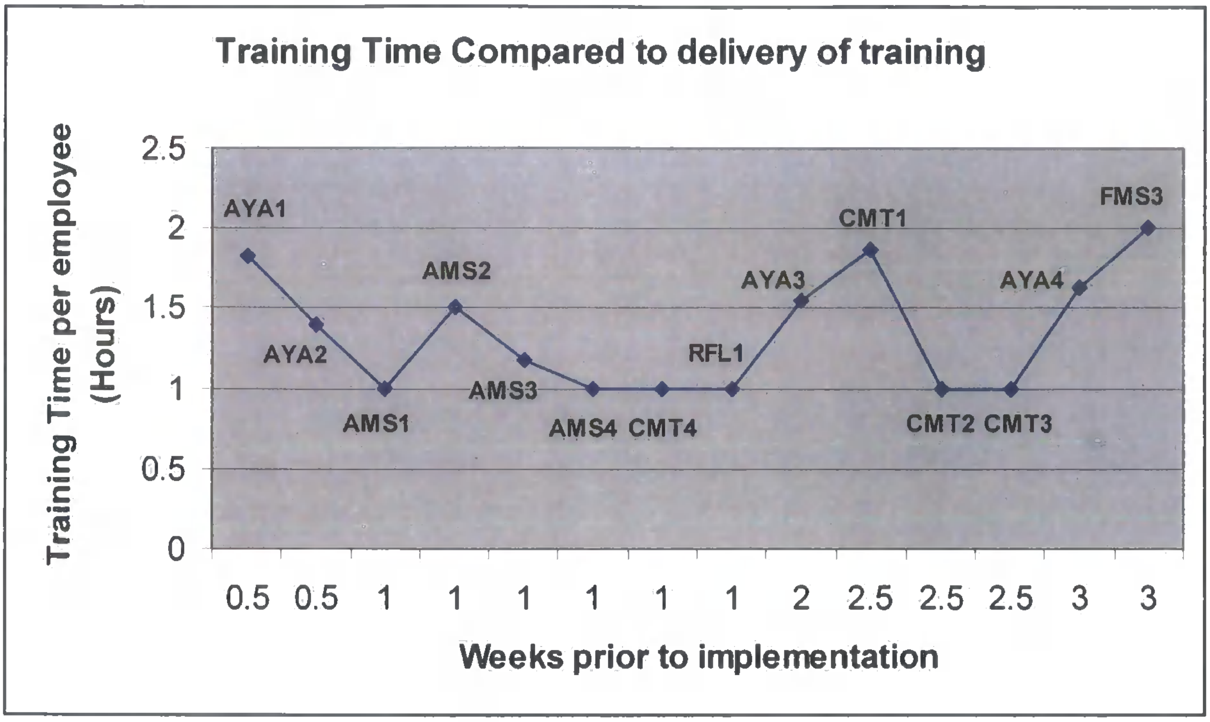


Figure 6-2 – Training time compared to delivery of training
Personality Mix Study

Figure 5-14 illustrates that 30% of end users were Pragmatists. An examination of the reasoning behind this result can be provided by analysis of the reasoning behind the MIS project. The MIS project was initiated to increase the standardisation of performance measurements across the production lines within the Philips factory. The work carried out within the TCS mini-project identified that many lines within the factory had differing methods of calculating performance measurements. Management were becoming increasingly concerned about the discrepancies and so the idea of the MIS came to fruition.

With reference to Figure 5-14 the following conclusions can be made:

- The majority of end users were quite sceptical about the introduction of the MIS and were only really willing to use the new application after other users had demonstrated its capabilities. The pie chart documented under the TCS mini-project results highlighted that the major concern of MIS operators was that data collection was far too time consuming. The high number of pragmatists within the operational end user community seems to stem from the concern that current data collection methods were taking up too much time and the introduction of a new method would only increase this time. However, once the success of the MIS had been proved these pragmatists became quite willing to accept the system.

- 40% of Late Adapters existed within the end user group. Before this result can be analysed, it is important to familiarise the reader with end users involved in this study. As documented previously, operational staff were highly concerned that current methods of data collection were not acceptable. However, the current methods had been in place within Philips Washington for many years and although time consuming, provided people with the information that was required. Martin **[MARTIN, 1991]**, states that the task of changing the culture of an organisation is a very difficult one, employees tend to be very reluctant to change the techniques that they have been using for years. With reference to Figure 5-14, the percentage of late adapters seemed to suggest that this was the case within Philips. Although employees knew that a change was needed, they were reluctant to move away from the tried and tested methods that they were currently using.

- RAD places a heavy emphasis on the end users and end user responsibilities are a lot higher than in SDMs of a more traditional nature. This heavy reliance on users makes the task of getting to know the end user group and their needs even more crucial.

- This study has helped to highlight that not only do the project requirements need clear definition; users personal requirements also need consideration. From the results identified it can be seen that in comparison to the model of a typical organisation provided by Martin in Chapter 3, Philips did possess a large percentage of late adapters. If this study had been carried out before project initiation more time could have been spent listening to users' needs, identifying their reluctance to change and addressing these problems before the system was designed. It is of little use delivering a system that is fully operational if the users feel that their personal needs have not been met.

The second 'additional' user analysis technique applied once the project was complete was an assessment of the end users motivation; using the results of the study the following analysis can be made:

- In Figure 5-16 examination of the operational end user community shows that one of the main motivational factors was that of reduction in workload. Operational users felt that the current methods of collecting and entering production information meant increases to their workload. Too much time was spent collecting performance information and not enough time was being dedicated to production. The next major motivator was that of how the new system could simplify their job. If a new system could reduce their workload whilst making their job easier this would help to increase the users' satisfaction with the system.

- Examination of the managerial end user community, Figure 5-16, illustrates that one of their main motivational factors was that of how accurate the MIS could make reporting of performance measurement. The second main motivator was that of the speed at which management could get access to data. If a system was produced that helped management to increase their data accuracy whilst at the same time increasing the speed at which they could access and report on data, this would satisfy their requirements and make them more motivated to use the system.

This study has helped to identify the key motivational factors from both end user communities in respect to the use of the MIS. By examining the factors that would 'drive' the users to use a new system this could help the development team get a better understanding of the users' needs. If this technique had have been used prior to initial project development, it could have provided the development team with a method of assessing key motivators. These key motivations could then have been taken into consideration when building the final system ensuring that the final product didn't just satisfy user requirements but also gave the user a system that they felt motivated to use. The final MIS that was implemented did satisfy the requirements specified by the user however in some cases because user motivation had not been examined the users felt compelled to have to use the system. This then lead to resentment of the system in some cases and made the management of change more difficult.

A further 'User Focussed' tool that was used to assess the user support provided by RAD was a user satisfaction survey. Approximately fifteen end users (both managerial and operational) were interviewed with respect to their satisfaction of the MIS system. Users were asked to rate the system into the following categories: Very satisfied, Satisfied, Dissatisfied, Very dissatisfied.

Figure 6-3 below illustrates that of the cross section of users interviewed 15% felt very satisfied with the system and 55% felt satisfied with the system.

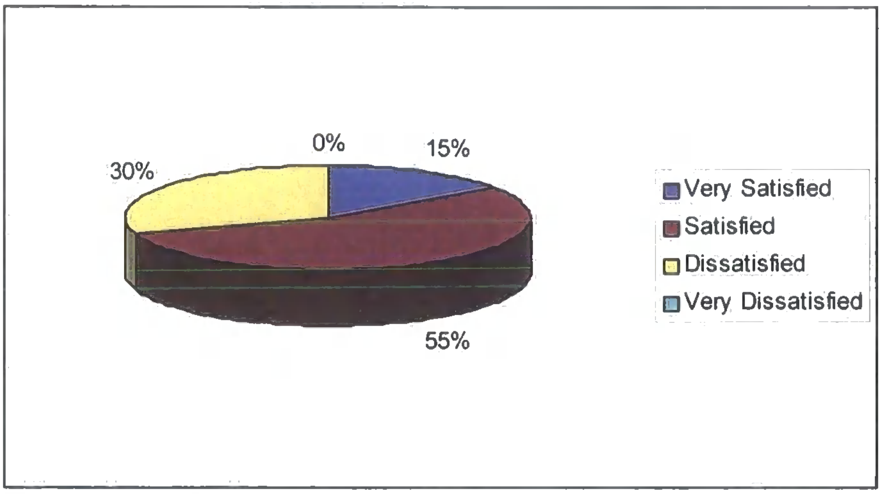


Figure 6-3 – User Satisfaction Survey

Of the 30% that stated they were dissatisfied with the system 20% of related problems stemmed from the flexibility of the system. Users explained that they were happy with the data being collected and reported on however they needed more access to design their own reports and publish their own graphs. This consideration will be taken into account in Phase 3 of the project. The remaining 10% of the concerns related to the speed of the system. Unfortunately due to the level of the data stored in the MIS response time is slow and is being examined with an aim to increase reporting time.

In summary, the results provided by analysis of RADs ability to provide increased user support have helped to identify any areas of concern within the end user group. If more of the techniques documented could have been carried out in conjunction with the TCS mini-project prior to initiation of the MIS project, it may have been possible to work more closely with the users and alleviate any fears that they were experiencing. In the long term, by increasing awareness of the users' needs through examination of the mix of personalities within the end user community, this may have helped increase user commitment to the project. This in turn could have reduced the development process in terms of training requirements and user sessions. This analysis has helped to highlight that within a RAD environment the need for increased user support, structured, well delivered training and generally high user involvement is key to the success of the project. It is important to note however that the above techniques are not exclusive to RAD and hence could be used to increase user support within alternative SDMs.

6.2.4 Frequent Delivery of End Products Through Prototypes

This section will analyse the results provided in Chapter 5 and examine the perceived benefit of the frequent delivery of end products through prototyping.

Once an initial set of requirements had been documented, through the URS and management meetings, a prototype was developed based on the identified requirements. The RAD prototyping method coupled with a Rapid Prototyping approach differed from traditional prototyping in that the prototype was used as a tool to be built upon and then implemented as the final system. With reference to Figure 5-25 it can be seen that the prototype was used in all stages of the development lifecycle with several releases of the prototype. Prototyping through RAD helped to provide the following benefits:

- An early prototype encouraged developers to satisfy the users needs, to interact with them and obtain their trust.
- It helped to ensure that requirements were continuously revisited and revised.
- With each release of the prototype, users could visually see their requirements being fulfilled, this helped to gain acceptance of the system.
- The prototype encouraged verbal communication, allowing developers to sit with users, talk them through the system and listen to their needs.
- Prototyping provided users with visibility of project development. Within the initial stages of the project users were unaware of what the role of the IT department was, they were so used to new systems 'appearing' on their desks, with no real interaction ever taking place with the IT department.
- Not only was prototyping useful for the users, developers also found the RAD prototype useful as it provided them with a base line for scope and clear requirements gathering.
- Prototypes were used to facilitate system testing. Usually in traditional methods a set of requirements is submitted, developers design and implement a system and then developers test it for functionality. RAD allowed the MIS development team to produce a fully operational prototype that the users could then test to see if it satisfied their requirements. Based on information taken from the issues database and MS project plans, Figure 6-5 below illustrates the average response time in solving issues.
- The prototype provided users with frequent releases of the proposed system. Based on information taken from the issues database, Figure 6-4 below illustrates the number of change requests made after major prototype releases.

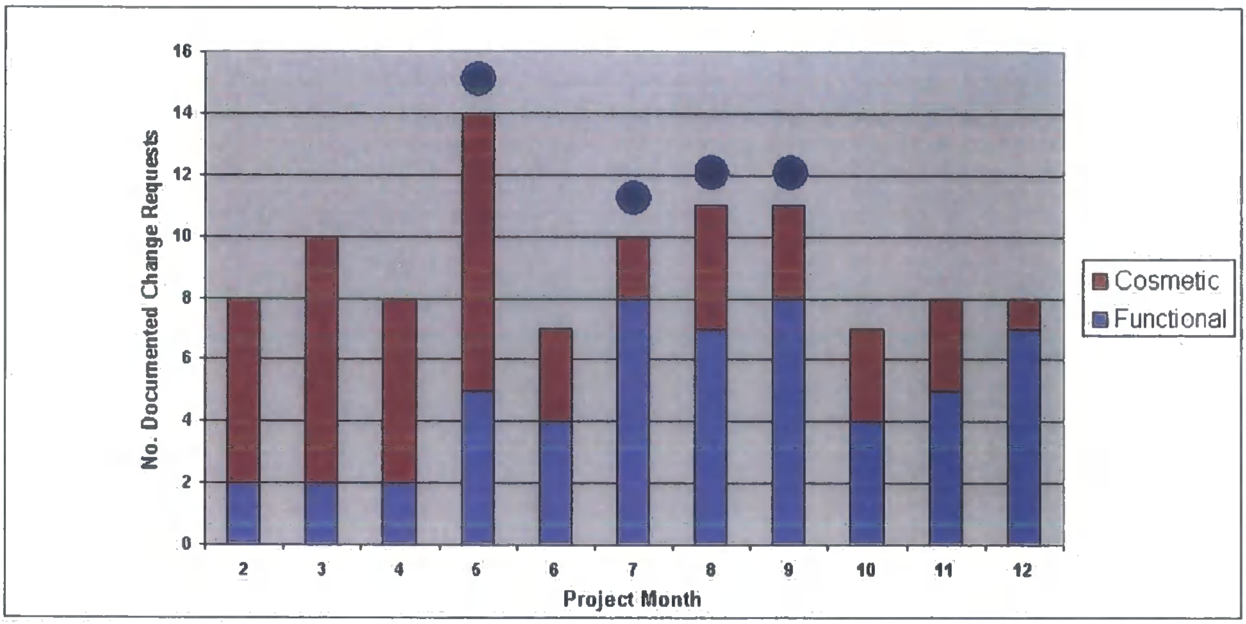


Figure 6-4 – Change Requests following major prototype releases

● Major Prototype Release

Generally it is expected that as a project progresses the number of functional change requests will reduce, however it can be seen from the above graph that this was not the case. The main reason behind this was the fact that two user groups existed and the increase in functional requests from month 11 onwards can be attributed to managerial requests as opposed to operational end user requests. The majority of these functional change requests were a result of changes to the KPI definitions requested by managerial users. Figure 6-5 below illustrates the average response time in solving user issues/requests.

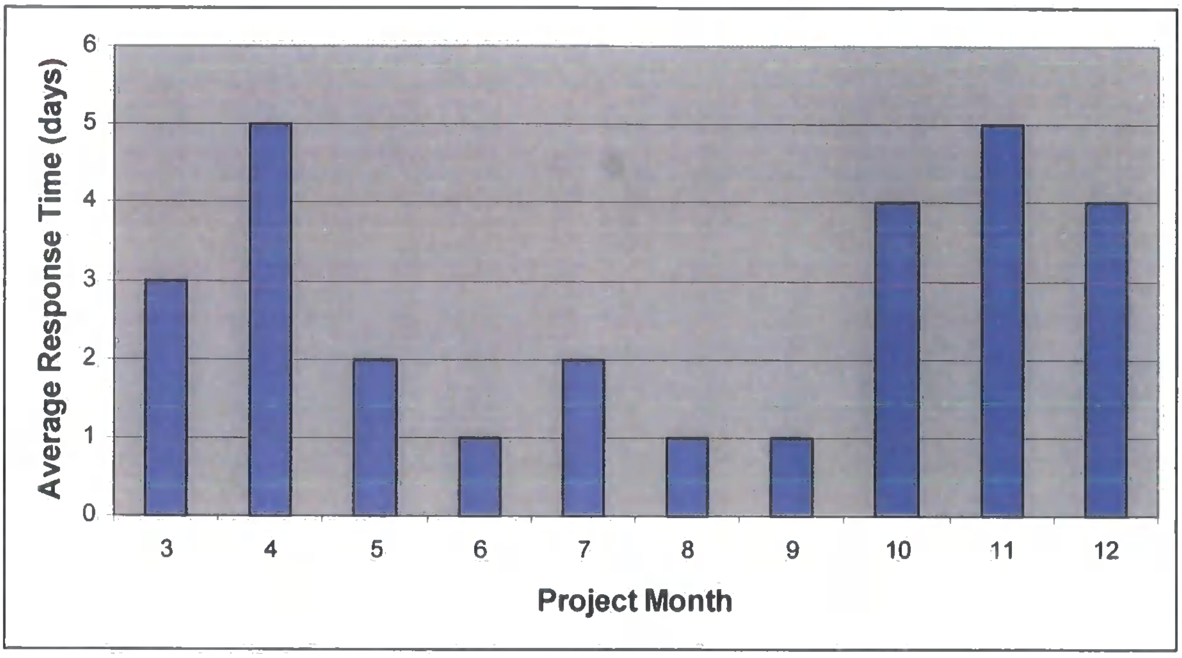


Figure 6-5 – Average Response Time in solving issues

As documented in Chapter 3, one of the benefits of the RAD approach is the ability to prototype user requirements at an early stage in the project lifecycle. With reference to Figure 5-25 this was true of the MIS project. Within three weeks of project start up some form of prototype had been released.

In summary it can be seen that RAD did help to provide frequent results through prototyping. One of the main reasons for this is the iterative nature to the RAD SDM. As with many of the perceived benefits of RAD, it is important to note that this iterative prototyping however is not exclusive to the RAD SDM. Frequent results may also be provided when using a similar SDM to that described in section 3.3.2.

6.2.5 Summary of 'Positive Factors'

From the above analysis it can be seen that in the majority of cases, the perceived benefits of the RAD SDM can rarely be assessed in isolation. As with any analysis, it is felt that an organisation wishing to apply RAD simply to provide a benefit such as reduced development time must also consider the wider implications. It is crucial that examination of all of the documented benefits be considered and an assessment of priority of applicability to business objectives carried out. RAD can provide many benefits but it is essential that it is applied for the right reasons.

In the case of the MIS project RAD did help to provide a successful solution. Products were delivered within the required timescales and user requirements were satisfied. In comparison to the following statistics taken from Computer Weekly publication, the MIS project can be categorised as a successful project.

According to Capers Jones **[JONES, 1999]**, large software systems run late and exceed their budgets more than 50% of the time. The primary reason why software projects run late is that half of the business year [of a major project] is spent on bug finding and fixing, not on development. A further contribution to project failure, is a conspicuous lack of application of formal metrics and project management tools during a project. Out of the 36 million software projects in the world in 1999, only 1% had effective measurements.

The figures below illustrate the findings by Jones regarding the associated statistics of "Best in Class", successful projects.

- Cost over-runs are less than 5% and schedule over-runs are less than 3%.
- Development productivity averages 50 function points – about 6,250 C statements per month, at less than £170 per function point.
- Reuse of design, code and test cases is more than 75% for all projects.
- Projects take less than nine months.
- Defects are fewer than 2.5% per function point, and removal efficiency is more than 97%.

- Delivered defects average less than 0.075% per function point.

With reference to the above the MIS project satisfies at least three of these criteria. The project was delivered on time, the initial MIS system was delivered within nine months, cost over-runs were less than 5% and reuse (once a configuration management tool had been applied) was around 75%. Hence from the Philips viewpoint this project can be classed as a successful project. However the MIS project is being assessed against 'typical' Philips projects that have been carried out in the past using standard methods of project planning. Hence these results may not be totally reliable for future projects if planning methods or project management methods change.

A further key lesson learnt is that RAD is not intended to be applied in a prescriptive manner. The techniques and methods documented as part of this study are intended only as guidelines and should be carried out at the appropriate phase of the development lifecycle. For example even though some of the user considerations such as 'personality mix' and 'motivation of users' had been overlooked at the beginning of the project, further analysis was carried out once the final product had been implemented. Although the results of these studies had no direct feed into the implementation of the initial MIS system, they proved very useful for subsequent phases of the project.

6.3 DISADVANTAGES OF THE RAD APPROACH

As with any SDM, RAD did demonstrate some of the perceived disadvantages to its approach. However in the case of the MIS project the majority of problems encountered were due to the fact that research into the RAD SDM was not fully complete before MIS development commenced. For this reason some of the well-documented pitfalls were overlooked and hence problems did occur. This section will use the results provided in Chapter 5 to analyse these.

6.3.1 A Change of Culture

One of the main perceived disadvantages of RAD are the changes imposed to the culture of a company. Martin states in Chapter 3 that the introduction of the RAD SDM can appear to be very threatening to employees unless changes are managed correctly.

In a company such as Philips it would be easy to understand the opposition of change due to the success of current working practises already in place throughout the factory. However this did not prove to be the case. Throughout the MIS project although a dedicated IS manager was not employed as part of the development team, the key project champions (initiators of the project) agreed that there was a need for a new method of working and hence their full support for the RAD SDM was given. The project champions agreed that due to the short timescales and the resources required for the project that RAD would be the most applicable. Due to empowerment and the high involvement of end users, the management of the changes imposed by RAD went reasonably smoothly. The majority of the end users welcomed the new approach of actually being involved in the development process as opposed to just being providers of information.

The key lesson learnt through the application of RAD is that user involvement and clearly structured user management is crucial to the success of change management. Involving users throughout all stages of the development lifecycle allowed the development team to identify if users had any concerns and if so address them accordingly. This ensured that concerns were not ignored such that they built up to a stage where they could not be managed. Having the full support of the project champions definitely helped this management of change.

6.3.2 Heavy People Focus

One of the quoted disadvantages and advantages of the RAD SDM is its heavy reliance on involvement of key personnel. As mentioned above since research into RAD was on going at the time of development, this key factor was overlooked and hence problems were encountered. The main reason why heavy reliance on users can be disadvantage occurs when assessing the availability of the required users. The application of RAD to the MIS project highlighted the need to ensure that key personnel are involved throughout the project not just the initial stages. The results provided by the assessment of management meetings and user sessions help to highlight this in Figures 5-6 and 5-7. The following observations can be made:

- Due to restricted availability of key personnel, once initial requirements had been discussed, the frequency of the meetings tended to drop to between two or three meetings a month. At a stage in the project where key personnel should have been involved constantly, the frequency of management meetings was reducing. The outcome of which meant that project decisions were being made without all key personnel being involved; this introduced some initial confusion surrounding the definition of KPI measurements, factors that were crucial to the operation of the MIS.

- Throughout the design of MIS with support tools such as the user guide there was little or no user input to the design. Again due to employee constraints and short timescales it was not viable to involve a user group in the design of support tools. In the case of the MIS project, user involvement in this area was overlooked mainly due to the fact that resources could not be spared for the job.

- Further examples of how the lack of committed resources affected the overall project outcome are the 'Personality Mix' study and the 'Motivation of users' study. Again due to the heavy reliance on users, these studies could not take place at time where they would have been most beneficial to the project. Ideally if time could have been devoted to work on these exercises before project initiation, a better understanding of the end user group would have been obtained.

Examination of the heavy reliance that RAD places on its user community has helped to identify a definite need to focus on user support tools well in advance of project implementation. In the case of the MIS project due to the tight timescales of development, this method of support tended to be slightly overlooked. If RAD were used in the future a suggestion would be to allow more time to examine the users support requirements in conjunction with RA. This would help to identify any issues and also provide the development team with a better understanding of how to meet the users needs.

6.3.3 Integration

A further perceived disadvantage of RAD is that of integration of the SDM into the current working practices employed by an organisation. Martin states that in many cases integration is often overlooked and that most organisations tend to use RAD for stand-alone systems without considering the other systems in place.

One of the key drivers behind the initiation of the MIS project was its ability to interface to several existing systems. As illustrated by the Stakeholder diagram in Chapter 2 it can be seen that the MIS needed to be designed with several other systems in mind. The success of the MIS project relied heavily upon existing sources of information around the factory. Due to the numerous information sources required, it was decided that the MIS system be built in several stages. The first stage of which, involved a link to existing production and personnel information. In the future, stage two will examine the dependency on WINTA (time and attendance system) and MFGPRO (resource planning) information. Finally stage three will concentrate on the MIS providing information to a larger executive information system known as EIS. Examination of the integration aspect of RAD has provided the following observations:

- One of the key lessons learnt when applying RAD was that of ensuring that consideration was also given to the hard and software used by existing systems, failure to do so in the case of the MIS resulted in a product that could not communicate with existing systems.

- It became clear that although an in-depth analysis of all inter-related systems had been carried out, more attention should have been devoted into examining the compatibility between systems. In the case of the MIS the system had been designed with consideration for the information that other systems stored but no consideration to the format of this data and the connectivity between systems. Due to the compatibility problems encountered, a large amount of re-work had to be carried out once the project went live, this lead to increased development times and increases in cost.

6.3.4 Creeping Functionality

A further documented disadvantage of the RAD SDM was the need to determine when enough requirements have been gathered and development can begin. Due to the iterative nature of RAD, it is all too easy to end up with a vast list of requirements in a very short space of time. Since requirements are often revisited more than once, new requirements stem from original requirements and soon the definition list is doubled. Reference to Figure 5-8 illustrates that even five months into the project, change requests were still being made and changes to the system were being implemented as the change requests were deemed crucial requirements for the MIS. As suggested in Chapter 3 in order to overcome this problem a structured method of timeboxing should have been used. Since research into the RAD SDM was carried out in parallel with MIS development, unfortunately the timebox method was not used within the MIS project. However with reference to Figure 6-6 below it can be seen that once the final system had been implemented the majority of the change requests tended to be more cosmetic in nature. These changes incurred less of an impact on project deliverables than functional change requests.

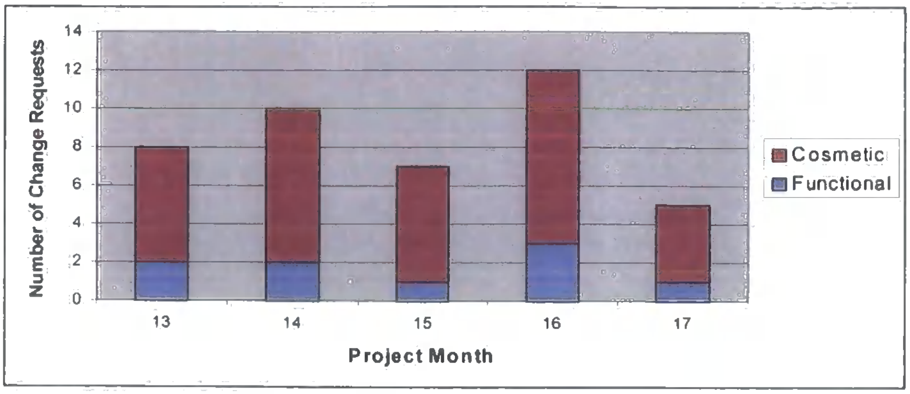


Figure 6-6 – Change Requests since implementation

When using the RAD SDM one of the main objectives is to gather as much information on user requirements as possible. In this respect functionality will grow as system development progresses. Throughout the MIS project since timeboxing was only really used in a reactive manner, it became very difficult to force a 'cut off' point to functionality changes. During the RA timebox, a set of mandatory and high priority requirements should be dealt with first and then any time left over at the end of the timebox should be used to add in some of the 'nice to have' functionality. If there are any lesser requirements that still need consideration, these should either be included in the next cycle or left as enhancement work at the end of the project **[DSDM, 1999]**.

6.3.5 Choosing the Right Tools

In Chapter 3 Chasan **[CHASAN, 1999]** states that often managers make the mistake of trying to economise and limit software purchases. She states that an effective tool can mean the difference between a dynamic, responsive RAD team and a stressed, ineffective team. Martin continues to state that within RAD, it is highly important to find a toolset that does what is required at the onset of a project rather than continuing with a mediocre toolset only to swap it half way through the project. Within the MIS project there are several examples of how the lack of research into the chosen tool hindered the progress of the project.

The MS Visual SourceSafe tool is one example of this. The following observation were made:

- The one big disadvantage with the use of this tool was the lack of technical understanding of the tool. Since the tool demonstrated all the capabilities that the development team required, it was decided that this was the best choice of tool without ever really considering any alternatives. However as the project progressed certain development issues arose and it became clear that the development team could not solve these issues due to lack of technical knowledge about the tool. It was therefore decided that the project progress without the use of MS Visual SourceSafe.

- A fair proportion of time was spent either recovering work that had been lost through Visual Source Safe, time which could have been saved if more time had been spent before project initiation researching the various version control tools available.

Although not necessarily deemed a 'development tool' as such, the MIS User Guide is an example of how the choice of support tool affected the project outcome. The following conclusions can be made:

- With reference to Figure 5-13 it can be seen that several re-issues of the user guide occurred, every time a change was made to the system, a new MIS user guide had to be issued.

- If more time had been devoted to choosing a 'better' tool with which to deliver the user guide a lot of time could have been saved in the re-issue of the document.

The most highlighted example of how the choice of tool chosen for the project affected the overall project outcome was that of the prototype tool. As documented in Chapter 5 unfortunately due to lack of skill sets and time available for tool research the MS Access application was chosen for the MIS prototype. The following observations were made:

- Access did provide a quick method of prototype development and was useful in providing an easy to use GUI (Graphical user Interface) tool.

- As the project progressed it became clear that the functional capabilities of MS Access were very limited, Access could not support the types of transactions required for the MIS

- As the project progressed and information storage increased the performance of the prototype became affected, transactions became slower and in some cases the database would fall over and cease to operate.

- Unfortunately due to the performance issues with the MS Access database a migration to an Oracle database had to be performed.

- Migration of all data and functions across to the Oracle environment had a dramatic impact on the development of the MIS. In total the migration process and associated support took approximately five months additional development work in order to produce a system that provided the same results as the existing MIS system.

- A further disadvantage encountered when using MS Access as a prototype tool was the amount of hard coding that had to be done in order to produce a visual representation of the system. Through MS Access there are very few visual tools that assist in prototype design, the interface relies heavily on programming effort.

6.3.6 Summary of 'Negative Factors'

As can be seen from the problems encountered above, the RAD SDM like many other SDMs does have some documented pitfalls to its approach. Within the MIS project two of these quoted disadvantages caused the majority of the problems; the fact that RADs heavy reliance on the user community was overlooked and the correct choice of tools used throughout the project. Due to these two problems, some knock on effects were encountered leading to the demonstration of further disadvantages.

The problems identified above however may have been avoided if research into the RAD SDM had been carried out independently and prior to project initiation. As illustrated in Chapter 3, the problems associated with the RAD approach are well documented and it is clearly suggested that a substantial amount of time is spent up front before project initiation, planning to deal with possible problems. The application of RAD within the MIS project demonstrated that the planning and support effort required before project initiation could take up as much time as the total support effort required once the project has commenced. Within the MIS project however, due to the fact that SDM research was carried out in parallel with development, this planning and support prior to project initiation did tend to be overlooked. This lack of forward planning then lead to further problems within the latter stages of the development lifecycle which could have been avoided if more time had been spent fully understanding the implications of RAD.

6.4 SDM Review

This section will provide a review on the applicability of the RAD SDM. Alternative, possibly more applicable SDMs will also be discussed.

The SDM chosen for the analysis of an IT system is largely determined by the nature of the system, no one SDM will suit all situations. Day **[DAY, 1999]**, in his Citielite monthly article states that there is no right or wrong choice of an SDM. He suggests that the following criteria be used when considering an SDM for a new project:

- Size of the development Team
- Time scale in which to work
- Skill set available

- Environment in which the software will be employed
- Style of management employed
- Nature of the system (Data vs. Process, as well as the rules imposed on it)

The above demonstrates the need to fully understand the situation under study before choosing an SDM.

For the purposes of the Philips MIS, prior to this study, the RAD SDM had been agreed as the chosen methodology primarily for its quoted fast delivery times. The results that this SDM provided are discussed in Chapter 5. However after performing research on alternative SDMs, in hindsight a more balanced assessment for the choice of SDM could have been taken. For example, using the criteria provided by Day **[DAY, 1999]**, one important consideration should have been the availability of skills within both the development teams and user groups. RAD places a heavy 'ownership' emphasis on both users and developers and as can be seen from the results provided, the majority of problems occurred due to lack of required skills or appropriate people to carry out the tasks.

After consideration of the above points a more suitable approach may have been to apply a combination of SDMs. For example the early modelling stages of SSM could have been used to provide a 'user' viewpoint, the results could then have been fed into the more structured waterfall SDM such as SSADM. Using a more structured approach like SSADM with its thorough documentation may have provided slightly more control to the project. However, I feel that an attempt to use all stages of SSADM may have lead to inaccuracies such as ill-defined requirements or requirements that no longer held true. The problem situation (see section 2.3) within Philips, although quite defined, still possessed some 'open-ended' or 'messy' human issues that were difficult to pin point. For this reason I feel that a full SSADM study would not have been appropriate as these human issues may have been overlooked or even ignored. By using the early modelling stages of SSM the human side of the problem situation could have been assessed in more detail thus providing a more thorough basis on which to enter the SSADM lifecycle. Once the Philips situation had been identified through SSM, the resultant output could then have been passed through to the feasibility stages of SSADM and then through to the problem solving stages such as selecting business systems options and technical systems options.

CHAPTER 7

CONCLUSIONS AND RECOMMENDATIONS

7.1 INTRODUCTION

This chapter will document the conclusions formed on the RAD SDM based on analysis documented in Chapter 6.

The conclusions documented in this chapter are based around the three project success criteria documented in section 1.3.

Examination of the perceived characteristics of the RAD approach.

Recommendations on how RAD could be tailored to overcome identified shortfalls.

Investigation of the applicability of the RAD approach when designing a software solution within a manufacturing environment.

7.2 MAJOR FINDINGS

This section will document the major findings of the study with respect to the RAD characteristics documented in Chapter 3.

One of the main results from the study was that as an IT project the MIS project was deemed a success. The two main contributions to this success were a product that was delivered within the agreed timescale and satisfied the users' requirements. The survey quoted in Chapter 3 carried out by Coopers and Lybrand stated that in 1996 eight countries including the UK concluded that two thirds of IT projects in the previous two years were late, cost more than the budget or failed to meet user requirements **[HOWARD, 1997]**. In this respect the MIS project was deemed a success. Several of RADs documented benefits also helped to contribute to this success, these will now be summarised.

- **Increased Product Quality** - The application of the RAD SDM did help to contribute towards increasing the quality of the delivered product. One of the main contributions to this increase in quality was the high level of user involvement throughout the RA stage of the project. This heavy user involvement ensured that the most current functionality was built into the system helping to provide a system that satisfied the users needs

- **Increased User Support** - The RAD SDM is intended as a flexible framework to be tailored to specific solutions and as such no set documentation exists with regards to user support as part of the approach. However, within the MIS project RAD allowed the development team to apply several user support tools such as User Guides and User Training. These techniques coupled with a high level of user involvement helped to ensure that the users received the required level of support and encouragement to use the delivered product.

7.3 Considerations To Be Aware Of

Although the project was deemed successful, there were some drawbacks associated with the RAD SDM. These will now be discussed.

- **Choice of development tool** - Throughout the MIS project several examples where the wrong choice of tool lead to problems were encountered. The heavy reliance that RAD places on the correct toolset coupled with the speed of development contributed to several failings such as increases in cost and resources required.

- **Overlooking heavy reliance on user community** – Although one of the main advantages of RAD is its heavy user focus, the availability of the right users must be thoroughly examined. Within the MIS little time was spent analysing the demands required from the user group, in some cases the demands for user involvement could not be met due to lack of available resources. This in turn affected decisions that were key to the success of the project and hence affected factors such as overall project cost. It was found that unless time is spent prior to project initiation, assessing the demands required of the user group, RAD could impose some fairly unrealistic demands on the involvement of end users.

- **Gaining Consensus** - A further consideration identified was that of the heavy focus that the RAD approach places on gaining consensus. Within the MIS study not enough emphasis was placed on gaining consensus at each stage within the development lifecycle. Often decisions were delayed until future phases by which time the impact on development imposed by changes sometimes became unmanageable.

- **Scope Creep** - The effective management of scope creep did prove to be a problem within the MIS project. The majority of difficulties were encountered once the prototyping stage began. Due to the iterative nature of prototyping the base functionality of the system soon began to grow and in some stages became difficult to control. As stated previously if the RAD SDM had been researched prior to project initiation, the timebox method would have been widely used throughout the project. Unfortunately since SDM research took place in parallel with development, the timebox method was only used reactively when problems arose.

- **Integration** - Throughout the MIS project, integration with other systems did tend to be overlooked due to the fast timescales within which the system was developed. Within the initial stages of the project little time was spent analysing the interdependencies between the delivered system and other existing systems. This lead to an MIS that was incompatible with existing systems.

7.4 RECOMMENDATIONS

Based on the application of the RAD SDM throughout the MIS project, the following section documents the recommendations for future RAD projects.

- **Assessment of skill set** – If RAD is to be used for future projects it is suggested that more time is spent prior to project initiation assessing both the skill sets and the software toolsets available within the organisation. Both of these factors did tend to be overlooked within the MIS project, tools were chosen based purely on the current skill set within the organisation. Although purchasing a new piece of software and training employees in its use initially involves a lot of time and expense these costs can often be recovered as the project progresses. In the case of the MIS project, due to incorrect tool choices approximately 70% of the original application had to be re-written with a new tool before it could be installed factory wide.

- **The correct choice of tools** - Although RAD documentation states that the choice of tools used within the project needs a great deal of consideration, it was felt that more focus could have been placed on the tools used throughout the prototyping stages. DSDM **[DSDM, 2000]** in their study of the RAD SDM state that the correct choice of both tools and techniques is crucial to the success of any project. The DSDM overview states that RAD tools and techniques must:

    - Enable rapid development

- Involve users in design and build
- Support iterative development
- Build excellent user interfaces

They state that it is not what you use that matters its how you use it that counts **[DSDM, 2000]**. If the RAD SDM is used in the future it is suggested that a greater percentage of time be spent prior to project initiation researching various prototyping tools and their relevance to the final product. It is suggested that the tool chosen is the same or at least generates the same executable code as the final software that is intended for the development of the end product. This will help to avoid any incompatibility problems and save time by not having to translate the prototype code into a new language.

- **Communication of RAD techniques** - A further recommendation for the future use of the RAD SDM would be to communicate RAD techniques to all users of the proposed product. In the case of the MIS project as is probably the case in the majority of software projects, the chosen SDM was only really communicated throughout the IT department and with those users who had some financial connection to the project. It was felt that if the end users of the proposed system had had a greater understanding of the SDM they would have appreciated the demands imposed on them slightly more, thus making the introduction of the new approach easier. As was the case however, once roles and responsibilities had been defined and the overall project timescales discussed, the users were quite willing to accept the SDM and a new method of working.

- **Constant involvement of key users** - In the case of the MIS study there were two main end customers and hence two end user groups to consider. In situations like this it is suggested that when using RAD, a high level of involvement from both sets of end user is required especially throughout the RA stage. Within the MIS project it became increasingly difficult to ensure that both sets of users were involved at the correct stages within the project lifecycle. If RAD is used in the future it is suggested that prior to project initiation a fair proportion of time is spent compiling a defined end user management structure. This structure would document content such as roles and responsibilities, end user availability and possibly high-level user deliverables.

- **Integration** - One suggestion in overcoming any integration problems would be to break projects down into smaller sub phases and concentrate on the inter system dependencies within each sub phase. This would help to ensure that a more detailed analysis of the links between systems could be carried out and hence ensure the final system is more compatible.

7.5 INVESTIGATION OF THE APPLICABILITY OF RAD WHEN DESIGNING A SOFTWARE SOLUTION WITHIN A MANUFACTURING ENVIRONMENT.

There were two main objectives for choosing to apply the RAD SDM for the development of the MIS. Firstly the speed at which RAD could help deliver an end product and secondly the flexibility of the SDM when dealing with rapidly changing environments.

The main objective for choosing RAD was its ability to produce results within a fast timescale. The manufacturing environment within which Philips sits is a very fast moving and dynamic one, which is rarely static for very long. When proposing new advances in technology in order for Philips to maintain a competitive advantage over rival companies there is a need to get these new products out to market as soon as possible and be one step ahead of the competition. Hence the requirement for fast results and an SDM that is flexible enough to support a rapidly changing business environment.

As documented in section 6.2.2 it can be seen that RAD did help to provide results within short timescales. Comparing the initial stages of the project when a more traditional SSADM based approach had been carried out, it can be seen that through the continuous re-visiting of user requirements, products were delivered to the customer in a shorter timescale. However the analysis carried out between the traditional and RAD SDMs within the RA stage can be slightly misleading, as there is usually a high level of change requests throughout the RA stage compared with any other stage within the project lifecycle. Unfortunately since a traditional approach was only used within the RA phase this was the only comparison that could be taken. In the case of the MIS study, the main contribution towards the reduction in project lifecycle was the phased approach of the SDM. Throughout the project, whenever requirements were identified they were categorised into highly important and highly urgent. Only those requirements that were deemed to be highly important and highly urgent would be implemented, any remaining requirements would be deferred to a subsequent phase. This helped to ensure that only the required functionality was delivered within the required timescales.

The second objective for using RAD was its flexibility to deal with change. Since the manufacturing environment is so fluid requirements soon change and systems have to be redesigned. The iterative nature of the RAD SDM was very suited to this type of environment as requirements could be revisited and revised at any stage within the development process. The D.U. market within which Philips operates does fluctuate frequently and even within a two-year timescale several new products were introduced and existing products phased out. These changes in environment brought with them changes in information requirements and hence amendments to the project requirements. It was demonstrated that RAD was flexible enough to handle these fluctuations since requirements were never frozen. RAD provided the required flexible framework within which to build the MIS.

7.6 Further Research

- **Identification of user group** - Due to the heavy emphasis that RAD places on the end user community it is suggested that further work could be carried out examining the characteristics of RAD when applied to various types of end user community. In the case of the MIS, two distinctly different user groups were involved and in some cases because of differing opinions, having two user groups did have a noticeable impact on RAD characteristics. A useful study would involve examination of requirements from each key user group individually. An assessment of the impact of RAD could then be made to each user group. In a real world situation however this type of analysis would prove to be incomplete, as ideally both sets of user requirements need to be considered at the same time.

- **Application of RAD to 'back-end' systems** - A further study that could be considered is the examination of how RAD can be applied to projects that have no user interface. The MIS project was heavily GUI focussed and hence the design had a high level of user input. Within Philips a large proportion of IT projects involve a heavy focus on 'back-end' design so it would be useful to examine RAD characteristics when applied to a database design for example. However if a study of this nature was carried out it would be quite difficult to evaluate the application of RAD due to the nature of 'back-end' design projects. Typically projects such as database design place a heavy reliance on the skill of the designer within the testing phase. Employees that know and understand the code that was written to produce the end product usually carry out testing. If RAD were applied to this type of project the skill of the end users would need to be taken into consideration, it would prove difficult to fully assess the impact of RAD if the users involved did not possess the required skills for the project. In this type of project it would probably be more beneficial to place more emphasis on the other perceived benefits of RAD such as speed of implementation without solely concentrating on the user focus aspect of the methodology.

7.7 SUMMARY

It can be seen from the above that RAD did help to contribute towards a successful project. However certain well documented characteristics of this SDM were overlooked and hence problems were encountered. One of the main recommendations for future use of the RAD SDM would be to place a heavy emphasis on understanding RADs user focus as this did prove to be both an asset and a hindrance within the MIS project.

APPENDIX A

INTERVIEW QUESTIONS

Interview Date: 22nd December 1998

Interview Time:

Person Interviewed:

Position: Manufacturing Manager

1. What are the most common problems relating to communication i.e. Time, incorrect information etc. Give percentages if possible.

2. What are the favoured methods of communication between Lines and Management in the factory? I.e. spreadsheets, paper form etc.

3. Do lines communicate effectively horizontally or is communication purely vertical.

4. Is there a set of clearly defined standards for communication i.e. flowcharts indicating where information comes from?

5. Does the current organisation structure facilitate effective communication or do you think it could be improved.

6. What techniques are in place to ensure communication is effective i.e. monthly meeting etc.

7. What do you perceive the main objectives of the MIS to be?

APPENDIX B

MIS TRAINING LOG

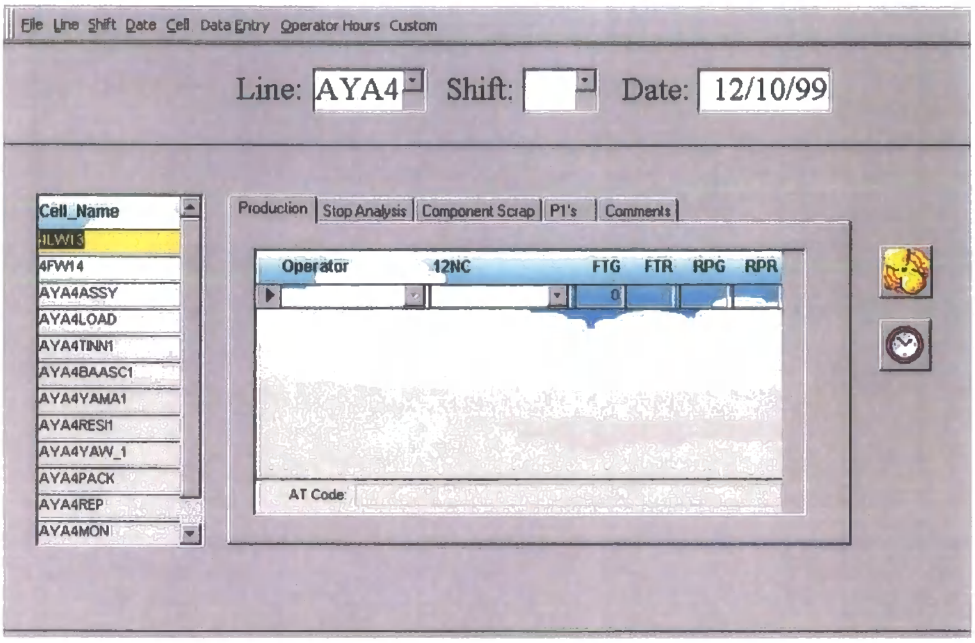
Line	A Shift	Hours	B Shift	Hours	C Shift	Hours	D Shift	Hours	WE Shift	Hours
AYA4	J. M P. R	3 2	J.D S.T J.W	 3	T.B B.C J. J	4 1				
AYA3	M. B P. F	1.5 2.5	L.H	1	B.E M.W	2 1	A.S	1.5	C. T I.A W. L	3 1 1
CMT 1	N. A N.W C. M	2 2 2	N. W H. G	1 1	S.M J. D	2 1				
CMT 2	C.H P. H	1 1	T. J	1	J. A	1				
CMT3	J.F L.W C. H	1 1 1	S. B	1	G. J M. C	1 1				
AMS 1	R. F	1	D.N	1.5	M. M	1				
AMS2	S. S	2	C. C	1	D. C L. R	2 1				
AMS3	R. P	1	P. D	2	M.B K. He	2 1				
					S. C	1				
					S. S	1				
FMS 3	S.J L. H	2.5 2.5	G.D L.H	3 1	L.S P. D	2 1	G. T S C L. D	3 3 3		
AYA1	T.G	1.5	A. H A.C	2 2	J.M	4	C.F	1		
AYA2	J.G	1.5	J. R	2	M.S	2.5	K.C	1		

APPENDIX C

MIS USER GUIDE

DATA ENTRY

To access the data entry screen the user must first enter their name and then press the Data Collection button on the 'Main Switchboard'.

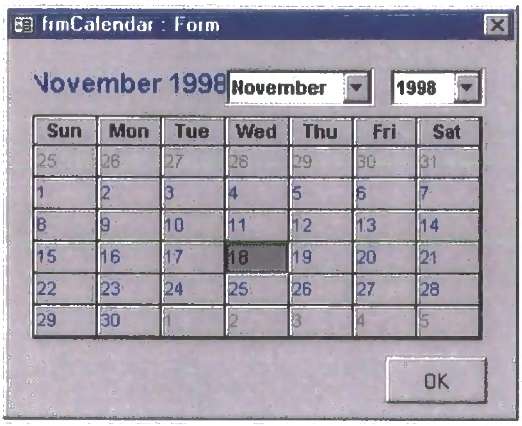


The above screen will be displayed. This is the main screen for day-to-day entry of MIS data. The main data entry section is split into five separate pages, a high level view of the required data will now be provided, a more detailed view is submitted in the following sections of this document.

Production page – this page holds information regarding production figures for each cell.
Stop Analysis Page – this page holds information regarding the downtime for a chosen cell.
Component Scrap Page – this page holds information regarding assembly scrap.
P1 Page – all winding P1 data is to be recorded on this page.
Comments Page – users can enter any comments regarding MIS data on this page.

HEADER INFORMATION

The 'Header Information' is information that is common to the records that the user is about to enter, namely, the Line, Shift and Date fields at the top of the data entry screen. The user may choose a Line and Shift from drop down lists. To change the date, the user must double click on the date field. When the user does this the following form appears:



Choose the date you wish to enter or examine data for by left-clicking it on the calendar. Then left-click the OK button. The Calendar screen will close and the date will be returned to the date field on the data collection screen.

APPENDIX D

CHANGE PROPOSAL FORM

URS Change Proposal Form

Report Title______________________________**Requested By** __________

Development Description (Brief):

Requested Date_____________ **Sign Off Date**______________

Required Equations For Output

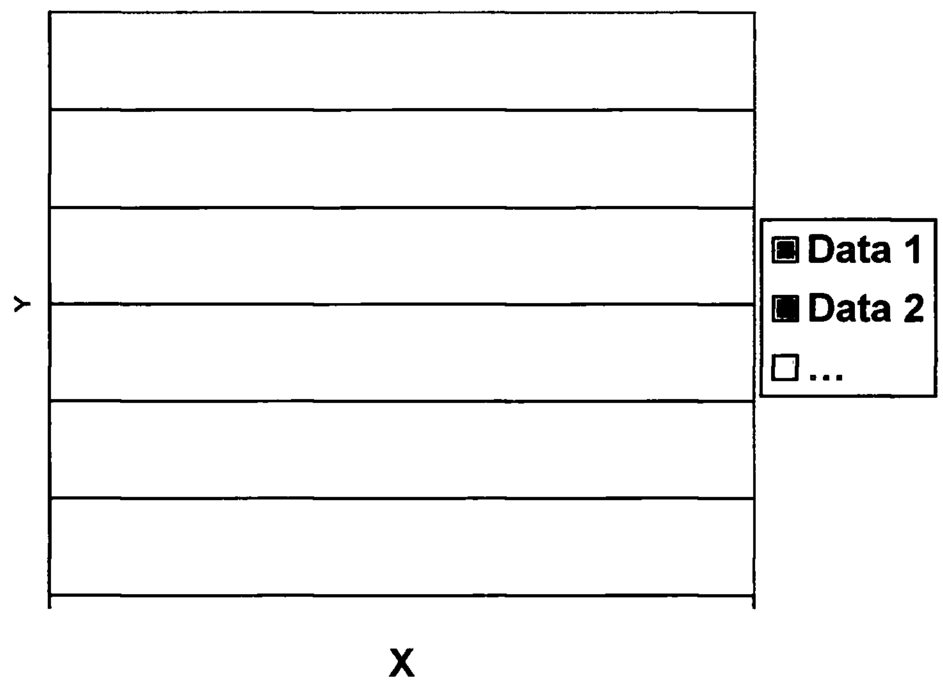
Additional Base Data (Definitions, Responsibility, Content, Timing)

Required Signatories: **Accepted By:**

Issued by: Name

Printed: dd/mm/yy hh:mm

Title



Filters

Line:
Shift:
Cell Name:

12NC:
Rejected due to:
Operator:

Component:
P1 Type:
Mandrel:

Between: dd/mm/yy and dd/mm/yy

APPENDIX E

MS PROJECT PLAN

ID	Task Name
1	**Requirements Elicitation**
2	Investigate current IS
3	Identify MIS Requirements (pre determined)
4	Investigate EIS
5	Prepare Data Requirements Report (includes data and manual form requirements)
6	**MIS Report Development**
7	Identify with all departments, what MIS reports they require.
8	Train departments to manipulate MIS data to produce their own reports (ongoing)
9	**Implement and Test Manual MIS system**
10	Implement MIS on test line
11	Refine MIS based on test implementation
12	Implement MIs on remaining lines
13	Refine system based on pilot test
14	**Migration of MIS**
15	Research appropriate Software Vendors
16	Investigate migration techniques
17	Implement Test Migration in parallel with current MIS across entire factory.
18	Migrate all MIS systems to new software
19	**Review MIS Database Design**
20	Investigate current Line Controller DB structure
21	Identify anonmolies in Line Controller vs MIS DB
22	Implement changes to MIs and Line Controller DB
23	**Integration Plan**
24	Develop Integration Plan for Introduction of automated MIS
25	**Implement and Test automated MIS system**
26	Investigate which areas to automate first
27	Automate MIS on test line
28	Refine automated MIS based on test implementation
29	Implement automated MIS on remaining lines
30	Refine automated system
31	Remove existing manual system
32	**Staff Training**
33	Prepare MIS Training Manual
34	Prepare training content
35	Deliver training
36	**Software maintenance strategy**
37	Develop and document strategy for future maintenance.

Project: 271098
Date: Wed 21/02/01

Task	Milestone	Rolled Up Task	Rolled Up Progress	Project Summary	Rolled Up Split
Progress	Summary	Rolled Up Milestone	External Tasks	Split	

APPENDIX F

ACRONYMS

Term	Meaning	Explanation
CASE	Computer Aided Systems Engineering	
CEO	Chief Executive Officer	
CPF	Change Proposal Form	
DB	Database	
DSDM	Dynamic Systems Development Methodology	
DU	Deflection Unit	
EIS	Executive Information System	Philips corporate information system.
ERP	Enterprise Resource Planning	
FTG	First Time Good	Refers to a good (non defective) DU that has been produced on the first production attempt.
FTR	First Time Reject	Refers to a bad (defective) DU that has been produced on the first production attempt.
GUI	Graphical User Interface	
IS	Information System	
IT	Information Technology	
JAD	Joint Application Design	
KPI	Key Performance Indicator	
LP	Language Prototype	
M	Managerial	
MFGPRO	Manufacturing Professional	Enterprise Resource Planning System.
MIS	Manufacturing Information System	
MRP	Materials Requirement Planning	
MS	Microsoft	
O	Operational	
OO	Object-Orientated	
PC	Personal Computer	
Philips Components		Refers to Philips Components Washington

Term	Meaning	Explanation
RA	Requirements Analysis	
RAD	Rapid Application Development	
RP	Rapid Prototyping	
RPG	Reprocess Good	Refers to a bad (defective) DU that has been adjusted to provide a good unit on the second production attempt.
RPR	Reprocess Reject	Refers to a bad (defective) DU that has been adjusted and is still a bad unit on the second production attempt.
SDM	Software Development Methodology	
SSADM	Structured Systems Analysis Design Methodology	
SSM	Soft Systems Methodology	
TOR	Terms of Reference	
URS	User Requirements Specification	A document containing the design specification for the required system.
VB	Visual Basic	
VSS	Visual SourceSafe	Microsoft Version control tool
WINTA	Windows Time and Attendance System	The clocking in system used by Philips.

APPENDIX G

BIBLIOGRAPHY

[**AVISON & FITZGERALD, 1995**] D.E. Avison, G. Fitzgerald, Information Systems Development, McGraw Hill, 1995.

[**BOAR, 1984**]. B. Boar, Application Prototyping: A Requirements Definition Strategy for the 80's, John Wiley: New York, 1984.

[**BOEHM, 1989**] B. W. Boehm, Software Risk Management, IEEE Computer Society Press, Washington, 1989.

[**BOEHM, 2000**] B. W. Boehm, Spiral Development: Experience, Principles, and Refinements, Spiral Development Workshop, February 9th, 2000.

[**BROCKWOOD, 1997**] T. Brockwood, A look at Microsoft Visual Basic and Layout from Objects inc, WEB Developers Journal, Markland Communities Inc, 1997.

[**BUCK-EMDEN, 2000**] R. Buck-Emden, The SAP R/3 System – An Introduction to ERP and business software technology, Addison-Wesley, 2000.

[**BUTLERGROUP, 1998**] ButlerGroup Consultancy, Managing Client/Server Development Presentation, 1998.

[**CARTER, WHYTE, BIRCHALL & SWATMAN, 1997**] Carter, C. J., Whyte, I. G., Birchall, S. A., Swatman, D. Rapid Integration and testing of business solutions, BT Technology Journal, Vol 15, July 1997, pp 37 - 45.

[**CHASAN, 1996**] G. Chasan, Trends in Rapid Application Development, Sybase Magazine, Q1, 1996, pp 46-49.

[**CHECKLAND & CHICHESTER, 1981**] P.Checkland, B. Chichester, Systems Thinking, systems practice, Wiley, 1981.

[CHECKLAND, 1990] P. Checkland, Towards coherent expression of systems ideas, Journal of Applied Systems Analysis 18; 25, 1990.

[DAY, 1999] H. Day, Strategies of Software Development, Citielite Monthly Article, Q1.

[CONNELL & SHAFER, 1989]. J.L. Connell, L.B Shafer, Structured Rapid Prototyping, Prentice Hall, Englewood cliffs, 1989.

[CSTB, 1990] Scaling up: a research agenda for software engineering. Communications of the ACM, 33, 281-293.

[CUTTS, 1987] G. Cutts, Structured Systems Analysis and Design Methodology, Paradigm, 1987.

[DAVIES, 1998] P. B. Davies, Rapid Application Development: an empirical review, Evolutionary Systems Development Lectures, Dec 1998.

[DSDM, 1999] DSDM Consortium, DSDM Method Overview, DSDM Consortium, 1999.

[DSDM, 2000] DSDM Consortium, DSDM The History of DSDM, DSDM Consortium, 2000.

[DE MONTFORT UNIVERSITY, 1999] Software Development Presentation, De Montfort University, 1999

[FITZGERALD, 1997]. B. Fitzgerald, The use of systems development methodologies in practice: a field study. Information Systems Journal 1997, 7, pp201-212.

[FOWLER & FOWLER, 1990] Fowler, H. W., Fowler, F. G., Clarendon Press. The Concise Oxford Dictionary of Current English, 1990.

[FOWLER, 2000] Fowler, C., Centre For Future Learning, Software Development Presentation, 2000

[FOUNDRY, 2000] Foundry OnLine, The Case for Rapid Prototyping, Foundry OnLine Web Site Report, 2000.

[FREDERICO, 1985] P.A. Frederico. Management Information Systems and Organisational Change. Praeger, 1995.

[GALLIERS & SUTHERLAND, 1991] R.D Galliers, A.R Sutherland, Information Systems Management and Strategy Formulation, Journal of Information Systems 1991,1(2), pp89-114.

[GRAHAM, 1998] I. Graham, Requirements Engineering and Rapid Development An Object-Orientated Approach, Addison-Wesley, 1998.

[GRINDLEY, 1995] K. Grindley, Managing IT at Board Level: The hidden Agenda Exposed. Second Edition. Pitman, 1995.

[GREENWOOD, 1974] W.T. Greenwood, Future Management Theory: A "comparative" evolution to a general theory. Academy of Management Journal, 1974, 17(3), pp503-13.

[HOWARD, 1997] Howard, A. A New RAD Based Approach to commercial Information Systems Development, Industrial Management & Data Systems, vol 5, 1997, pp 157 - 177.

[ISENSEE & RUDD, 1996] S.Isensee, J. Rudd, The Art Of Rapid Prototyping, International Thompson Computer Press, 1996.

[JONES & KING, 1998] T. Jones, S.F. King, Flexible Systems for Changing Organisations: implementing RAD, European Journal of Information Systems 1998, Part 7,pp 61 -73.

[KENDAL, 1989] P.A. Kendal, An Introduction to Systems Analysis And Design – A Structured Approach, WCB Publishers, USA, 1989.

[KINMOND & BHARTI, 2000] R.M. Kinmond, J. Bharti, Rapid Application Development Presentation, March 2000.

[LAUDON & LAUDON, 1999] K.C Laudon, J.P Laudon, Essentials of Management Information Systems,Prentice-Hall,1999.

[LAWS, 1996] A. Laws, Systems Analysis and Design Lecture Notes, Semester 1, 1996.

[MACLEAN,STEPNEY, 1994] R. Maclean, S. Stepney, S. Smith, N. Tordoff, D. Gradwell, T. Hoverd, S. Katz, Analysing Systems, Prentice Hall, 1994.

[MARTIN, 1991]. J. Martin, Rapid Application Development, Macmillan Publishing, 1991.

[McLEAN, 1973] E. McLean, Comment on Empirical Studies of Management Information Systems. Van Horn, p181.

[MULLIN, 1990] M. Mullin, Rapid Prototyping for Object-Orientated Systems. Reading, MA: Addison Wesley, 1990.

[MURDICK & MUNSON, 1986] R.G. Murdick, J.C. Munson, MIS Concepts and Design, Prentice-Hall International, Inc. 1986.

[MILES & SNOW, 1986] R.E Miles, C.C. Snow, Organizations: New concepts for New Forms, California Management Review Summer 1986, pp21-31.

[MILLINGTON & STAPLETON, 1995] D. Millington, J. Stapleton, Developing a RAD standard, IEEE Software September 1995, pp54-55.

[PAGE-JONES, 1980] M.P. Jones, The Practical Guide to Structured Systems Design, New York: Yourdon Press, 1980.

[PHILIPS, 1997] Philips Components Washington, Minutes of MIS Management Meeting, Q3, 1997.

[PHILIPS, 1998] Philips Components Washington, An Introduction to Philips Components, Induction Pack, 1998.

[PTAK & SCHRAGENHEIM, 2000] C.A. Ptak, E. Schragenheim, ERP Tools, Techniques and Applications for Integrating the Supply Chain, CRC Press, 2000.

[RAMAMOORTHY, GARG & PRAKASH, 1986]. C.V. Ramamoorthy, V. Garg, A. Prakash, Programming in the large. IEEE Transactions on Software Engineering 12 (7): 769-783, 1986.

[RIDGWAY-DAVIES, 1996] Ridgway-Davies, Managing Dynamic Change in a RAD Project, BCS Data Management Specialist Group Presentation, 1996.

[SMILES, 2000] Smiles, D. Software Testing and Software Development Lifecycles, Overview Report, Oct 2000.

[STAPLETON, 1995] Stapleton, J. A Quality Approach to Rapid Application Development, SQM, issue 25, 1995.

[UCD, 1997] University of California, Introduction to the RAD Methodology, UCD ADM Homepage, Systems Development Education Site, University of California Davis Campus, 1997

[WILSON, 1992] B. Wilson, Systems: Concepts, Methodologies and Applications, John Wiley & Sons, 1992.

[WILSON, 1999] B. Wilson, Brian Wilson & Associates Management Consultancy, What is Soft Systems Methodology Presentation, December 1999.

[YOURDON, 1982] E. Yourdon. Managing The System Lifecycle, A Software Development Methodology, Yourdon Press, 1982.